## DOCUMENT RESUME

ED 156 160 IR 006 142

| | |
|---|---|
| AUTHOR | Christensen, Don A., Ed. |
| TITLE | Proceedings of 1977 Conference on Computers in the Undergraduate Curricula. |
| INSTITUTION | Michigan State Univ., East Lansing. |
| PUB DATE | Jun 77 |
| NOTE | 354p.; For related documents, see IR 006 423-449 ; Parts of the document may be marginally legible |
| AVAILABLE FROM | Ted Sjoerdsma, 128 F ICM, University of Iowa, Iowa City, Iowa 52242 ($10.00) |
| EDRS PRICE | MF-$0.83 Plus Postage. HC Not Available from EDRS. |
| DESCRIPTORS | *College Curriculum; *Computer Assisted Instruction; Computer Graphics; Computer Oriented Programs; *Conference Reports; *Higher Education; Simulation; Symposia; *Undergraduate Study |
| IDENTIFIERS | *Minicomputers |

ABSTRACT

The proceedings of the Eighth Conference on Computers in Undergraduate Curricula include 54 technical reports on the use of computer technology in various areas of undergraduate curricula, including the biological sciences, business, economics, education, humanities, mathematics, physics, psychology, chemistry, engineering, social science, and statistics. Twelve additional reports are concerned with computer graphics, simulation, and use of the computer in testing. A computer aid for record keeping and two college programs are also described in individual papers. (CMV)

***********************************************************************
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
***********************************************************************

ED156160

U S DEPARTMENT OF HEALTH.
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

# Proceedings of
# 1977
# CONFERENCE ON COMPUTERS
# in the
# Undergraduate Curricula
# CCUC8

"PERMISSION TO REPRODUCE THIS MATERIAL IN **MICROFICHE ONLY** HAS BEEN GRANTED BY

Ted Sjoerdsma

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

*Edited by*
Don A. Christensen

*Hosted by*
Michigan State University

June 20, 21 & 22, 1977

IR006142

**Gerard P. Weeg 1927 - 1977**

Gerard Weeg, a major contributor to the initiation and success of the Conference on Computers in Undergraduate Curricula, died in Iowa City, April 10, 1977. As a faculty member, computer center director, department chairman, researcher, contributor in many fields and friend to all who knew him, Gerry Weeg will be sadly missed. We were privileged to have known and worked with him, and in his memory, CCUC/8 is dedicated.

Printed in the United States of America

Library of Congress Card Number 74-10711

Conference on Computers in the Undergraduate Curricula.
June 20, 21, & 22, 1977, Michigan State University
East Lansing, Michigan
Proceedings

College of Business/Michigan State University

Copyright 1977 : CCUC

## STEERING COMMITTEE

Gerald L. Engel
Virginia Institute of Marine Science

John W. Hamblen
University of Missouri at Rolla

Harry G. Hedges
Michigan State University

James A. Higgins
State University of New York at Buffalo

Alexander A. J. Hoffman
Texas Christian University

Glenn R. Ingram
U.S. Energy Research and
Development Administration

Gerald E. St. Amand
Michigan State University

Theodore J. Sjoerdsma
University of Iowa

David L. Stonehill
State University of New York at Binghamton

## LOCAL COORDINATING COMMITTEE

| | |
|---|---|
| Co-Chairmen | Harry G. Hedges<br>Gerald E. St. Amand |
| Coordinator | Raymond W. Smith |
| Conference Consultant | Kathy Stopa |
| Family Activities, Hospitality | Mara Alexander |
| Housing and Food Service | William G. Drake, Jr.<br>Donald R. McMillan<br>Robert R. Sutton |
| Publications Editor | Don A. Christensen |

The Eighth Conference on Computers in Undergraduate Curricula wishes to thank the following people for their contribution, effort, time, and knowledge as referees of the papers submitted for presentation.

Ronald Anderson, University of Minnesota
Richard Austing, University of Maryland
Robert Banaugh, University of Montana
G. Robert Boynton, University of Iowa
Mary Jane Brannon, Huntingdon College
W. J. Bulgren, University of Kansas
Tom W. Carroll, Michigan State University
Phillip L. Carter, Michigan State University
Frank W. Connolly, Montgomery College
Joyce Corrington, Xavier University of Louisiana
M. K. Cueman, Virginia Institute of Marine Science
Frank Degges, Virginia Institute of Marine Science
William Dorn, University of Denver
G. B. Dunn, Fairmont State College
Gerald L. Engel, Virginia Institute of Marine Science
Selby Evans, Texas Christian University
F. T. Fink, Michigan State University
Armand J. Galfo, College of William and Mary
John W. Hamblen, University of Missouri, Rolla
Carl Hacker, University of Texas Health Science Center, Houston
Harry G. Hedges, Michigan State University
James A. Higgins, SUNY at Buffalo
A. A. J. Hoffman, Texas Christian University
Glenn R. Ingram, U.S. Energy Research and Development Administration
Larry Jehn, University of Dayton
Baskeer Khumawala, University of North Carolina
Wayne Lang, MacMurray College
Martin Levin, Emory University
Joyce C. Little, Community College of Baltimore
Donald H. McClain, University of Iowa
Donald McLaughlin, Augustana College
Edmund Meyers, The Boys Town Center for theStudy of Youth Development
Andrew Molnar, National Science Foundation
Robert Montgomery, North Carolina Central University
Bruce E. Norcross, SUNY at Binghamton
Anne Permaloff, Auburn University-Montgomery
Ernest Pope, Palo Alto Unified School District
Leighton A. Price, Michigan State University
C. A. Quarles, Texas Christian University
Joseph Raben, Queens College of CUNY
R. C. Rosenberg, Michigan State University
Gerald E. St. Amand, Michigan State University
Theodore J. Sjoerdsma, University of Iowa
David Smith, Duke University
John B. Smith, Pennsylvania State University
Phillip F. Spelt, Wabash College

Ronald Stiff, Illinois Institute of Technology
David L. Stonehill, SUNY at Binghamton
Anna Marie Thames, Golden West College
John L. Van Iwaarden, Hope College
William Viavant, University of Utah
Rufus Walker, Jr., Louisiana State University, Shreveport
David Weinman, Hollins College
T. C. Willoughby, Ball State University
Frank Wojcik, Virginia Institute of Marine Science
Allen D. Ziebur, SUNY at Binghamton

We would also like to thank Marjorie Overhouse, Robert Weber and Karen Hudson who played important roles in completing this project on time.

Our special thanks to the authors whose cooperation kept this publication on schedule.

Don A. Christensen

# TABLE OF CONTENTS

BIOLOGICAL SCIENCES

Plants and Human Affairs: Educational Enhancement Via a Computer. . . . . . . . . . . . . . 1
*Theodore J. Corvello, W. Nelson Smith*

An Interactive Program for Carbon Labeling in Photosynthesis Written in ASCII PLI . . . 7
*Roy E. Gereau*

Teaching Mendelian Genetics with the Computer. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
*James W. Small, Jr.*

BUSINESS

Multimedia Approach to Business Data Processing Instruction . . . . . . . . . . . . . . . . . . . . 19
*Joel D. Stutz, Ronald D. Armstrong*

Use of a Computerized Forecasting Model in the Teaching of Basic Business Statistics. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
*Cecil H. Meyer*

The Lease/Purchase Decision . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 33
*H. Gary Kuhlmann*

ECONOMICS AND BUSINESS

Integrating Computer Concepts into Principles of Accounting . . . . . . . . . . . . . . . . . . . . 37
*Roy James Parrish, Jr., Henry J. Beck*

Computer Assisted Learning—Application to Planning in Command Economies . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
*William L. Henderson*

Use of a Deterministic Macroeconomic Computer Model as a Teaching Aid in Economic Statistics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 63
*John R. Tedford*

EDUCATION

Discover: A Computerized Careers Curriculum . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 71
*Jack R. Rayman*

ICS: An Intelligent Vocational Guidance Counseling System. . . . . . . . . . . . . . . . . . . . . 79
*Michael C. Hinckley, Paul Lorton, Jr., Robert L. Smith*

Computers and Teacher-Education . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
*Michael A. Lorber, Steven J. Skinner*

GENERAL TOPICS

Report on a Highly Used Computer Aid for the Professor in his Grade and Record Keeping Tasks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 93
*Richard Brockmeier*

Raising the Computer Consciousness at Skidmore College . . . . . . . . . . 101
*Margaret K. Brown*

Computer Assisted Instruction at Bennett College . . . . . . . . . . 105
*Nellouise D. Watkins*

GRAPHICS

Interactive Graphics with a Microprocessor Terminal . . . . . . . . . . 109
*John H. Karl*

Use of Computer Graphics in the Physics Laboratory at Washtenaw Community College . . . . . . . . . . 115
*George H. Kapp, Dwight D. Hinds*

Non-Interactive Graphics Techniques in Mathematics Education . . . . . . . . . . 119
*Kenneth Joy, James R. Warner, Richard Yusakaitis*

HUMANITIES

Computer Art—A New Course Offering for General Education . . . . . . . . . . 129
*Grace C. Hertlein*

Computing in the Humanities: A Bicentennial Experiment . . . . . . . . . . 145
*Nell Dale*

Umlaut β, and Learning German Verb Conjugation by Computer . . . . . . . . . . 149
*L. Allen Viehmeyer*

MATHEMATICS

Using Networks to Net Results . . . . . . . . . . 153
*John P. Tucciarone*

A Design for Computational Teaching Packages . . . . . . . . . . 159
*W. Y. Arms*

On the Development of a Proof . . . . . . . . . . 165
*Marialuisa N. McAllister*

Computer Oriented Calculus Courses Using Finite Differences . . . . . . . . . . 167
*Sheldon P. Gordon*

A Method for the Use of Computing in a Conventional Calculus Course . . . . . . . . . . 173
*Dennis D. Berkey*

Calculus and Linear Algebra in A.P.L. . . . . . . . . . . 179
*James W. England*

Computerized Applications of Matrix Algebra . . . . . . . . . . 183
*James T. Johnson*

Teaching a Survey of Mathematics for College Students Using A Programming Language . . . . . . . . . . 187
*Edward J. LeCuyer, Jr.*

Computers and Calculators: A Strategy for Combined Usage in Trigonometry . . . . . . . . . . 193
*Samuel W. Spero*

PHYSICS

Computer Exercises in Geophysics . . . . . 203
*Paul M. Doherty*

SPACETIME—An Experimental Learner—Controlled Dialog . . . . . 207
*Alfred Bork*

Course Management System for Physics III at Irvine . . . . . 213
*Alfred Bork, Jeff East, Joseph Marasco*

PSYCHOLOGY

The Role of Computer Supported Instruction in a Course on the Psychology of Oneself . . . . . 223
*Eileen G. Thompson*

Computer-Managed Instruction Produces Better Learning in an Introductory Psychology Course . . . . . 229
*James H. Roll, Robert M. Pasen*

Automated Administration of Psychological Instruments: Pedagogical and Practical Considerations . . . . . 239
*Leighton A. Price*

SCIENCE AND ENGINEERING

An Elective Laboratory Course in Minicomputers in Chemistry . . . . . 247
*P. E. Sturrock, G. E. O'Brien, A. C. Hayman, M. A. Goldstein*

Using APL to Facilitate a Self-Paced Course in Cost and Optimization Engineering . . . . . 257
*George N. DePuy*

TTP—A Computer Package for Teaching Transportation Systems Analysis . . . . . 263
*Mark Abkowitz, Steven R. Lerman*

SIMULATION

Teaching Basic Computer Simulation Techniques to Undergraduate Biological Science Majors . . . . . 271
*J. D. Spain*

Enriched Understanding of Simulation by Usage and Dissection of Pass IV . . . . . 275
*Warren Thomas, Harvey Axlerod*

Spatial Marketing Simulation, A New Package . . . . . 279
*Robert S. Ellinger*

Simulated Study of Organizational Theory: The Use of EXPER-SIM in the Undergraduate Business Curriculum . . . . . 287
*Paul Lorton, Jr.*

Using Macroeconomic Simulations in the Principles Course for the First Time . . . . . 295
*David S. Moewes*

Computerized Gaming and Simulation in Higher Education: The METRO-APEX Experience . . . . . 301
*Jane Wolin*

SOCIAL SCIENCES

From Manos to Matrices: Archaeological Classification and Computer-Assisted Cluster Analysis . . . . . . . . . . 307
*Douglas Caulkins*

Experimental Simulation in Introductory Sociology: Modifying EXPER-SIM to Meet the Needs of Beginners . . . . . . . . . . 313
*George H. Conklin*

Effectively Teaching Undergraduate Social Research Via Computer Data Analysis . . . . . . 323
*William R. Brown, Ida Cook, Charles M. Unkovic*

STATISTICS

Integrating Prepackaged Computer Programs into an Undergraduate Introductory Statistics Course . . . . . . . . . . 331
*Frank Scalzo, Rowland Hughes*

A Computer-Based Laboratory for Mathematical Statistics and Probability . . . . . . . . . . . 339
*Elliot A. Tanis*

The Computer as an Aid for Modular Instruction in Statistics . . . . . . . . . . 347
*William E. Geeslin*

TESTING

Effect of Computer-Delivered Testing on Achievement in a Mastery Learning Course of Study with Partial Scoring and Variable Pacing . . . . . . . . . . 351
*Richard M. Evans, Alvin J. Surkan*

Maximizing the Transferability and Utility of a Computer-Assisted Test Construction System . . . . . . . . . . 357
*P. L. Olympia, Jr.*

Computer Processed Evaluation . . . . . . . . . . 361
*George H. Griswold, George H. Kapp*

# Plants and Human Affairs: Educational Enhancement Via a Computer

**Theodore J. Crovello[1] and W. Nelson Smith**
*Department of Biology*
*University of Notre Dame*
*Notre Dame, Indiana 46556*
*(219) 283-7496*

### Introduction

Use of computers in undergraduate biological teaching is in its log phase of growth. Indications of this include increased participation of computers in biological teaching symposia held at the annual meetings of the American Institute of Biological Sciences, as well as growing numbers of articles on computers in biological teaching appearing in such publications as *Bioscience* and the *AIBS Education Review.* Another sign of increasing activity is the growing number of programs that deal with specific biological phenomena. Biological programs are available in large computing networks such as PLATO, from organizations such as CONDUIT, or from individual professors. This current increase in use of computers in undergraduate teaching also is evident from the results of a recent CONDUIT survey of 3,500 life sciences departments. This survey, overseen by Crovello, indicated a diverse level and type of instructional computing guidelines for their integrated use in biological teaching throughout an entire life sciences department. At Notre Dame, computers have been used in biological teaching at the undergraduate level in courses in general biology, ecology, genetics, physiology, plant taxonomy, ecology, and in plants and human affairs. The purpose of this paper is to describe our actual experience with the use of computers in one of these courses, "Plants and Human Affairs." We believe that its use has enhanced both teaching and learning, and that our innovative uses might be of value elsewhere.

### Plants and Human Affairs and the Computer Facilities It Uses

Plants and Human Affairs is a university-wide elective course (one open to students in every major). The kind of credit a student receives depends on the department from which he/she comes. With the exception that students must be juniors or seniors, there are no prerequisite courses or experience, either in biology or in computing. Enrollment in Plants and Human Affairs varies between 65 and 100 students each spring semester. It is not a required course for anyone, although it is a suggested elective for the Education for World Justice option at Notre Dame. The purpose of this lecture-only course may be summarized (perhaps too tersely) as follows: without plants we wouldn't be, and without plants we wouldn't be as happy.

Notre Dame's 9,000 students are served by the University's IBM 370/158 computing facility. It permits simultaneous use of forty-five remote terminals in addition to a batch processing background. The biology department has six terminals located in its new Life Sciences Building, but these are not used for Plants and Human Affairs for several reasons, including: insufficient availability of an adequate number of terminals; unavailability of the terminals in the biology department for extended hours throughout the day and night; and lack of funds to retain adequate supervisory personnel.

Plants and Human Affairs uses the IBM 370 exclusively. The department's own Tektronics 4051 and its Nova 1200, even though the latter has a time-sharing option available, are not used in Plants and Human Affairs, primarily because the department cannot supply a sufficient number of dedicated terminals required for such a large class enrollment.

## Types of Computer Use in Plants and Human Affairs

The authors' educational philosophy centers on the concept that the wise use of computers can enhance undergraduate teaching and learning. It has been our policy that when some use does not accomplish this, it is discontinued. In addition, we believe that almost every undergraduate course can benefit from some amount of computer usage, even if it is only to permit more efficient information retrieval from the literature. Arising almost as a corollary of this philosophy is our commitment to reduce the feeling among undergraduate students that they are simply passive parasites on the educational system. Our major approach to alleviate this feeling has been to encourage students to create computer programs which would be useful in subsequent offerings of undergraduate courses, and also to create the actual biological questions, answers, and replies to answers which are used in on-line multiple choice course reviews. This will be described in more detail later.

Computers can be used in undergraduate teaching in seven ways: information retrieval, automated data accumulation, data analysis, simulation of biological phenomena, decision making, online control of experiments, and multiple choice course review. Plants and Human Affairs uses computers in three of these ways: information retrieval, multiple choice course review, and simulation. We now shall discuss each of these three uses in greater detail.

### Information Retrieval

As its title implies, Plants and Human Affairs covers a huge spectrum of knowledge, from cellular biology and plant anatomy up through world economics and environmental quality as they are affected by plants. One of a professor's biggest problems in such a course is to remain up-to-date with the literature. This is not just the problem of keeping informed as to what has been published, but also of keeping informed as to what references and information he has already added to his files. Consequently, several years ago Crovello created a personal computerized data bank of literature references of interest to him in either teaching or research. Each reference is deep-indexed, and the names of the authors, date, title, citation, and deep-indexed subject numbers are entered into the computer data bank. Any combination of key words and subject numbers can be searched for with a simple descriptive command. For example, to provide the lecturer with all references on plants and poisons, the subject number for plant, for poisons, and the Boolean logic connector are all that must be entered onto a punched card for computer searching. To be extra certain of not missing any relevant references in the data bank, the subject words "poisoned," etc., which might appear in certain titles, also could be looked for. Currently, Crovello's personal literature bank contains about 7,500 references.

Because Plants and Human Affairs covers so much material, and many students are seeing such material for the first time, heavy reliance is made on visual aides, particularly two by two projector slides. In a manner similar to literature references, Crovello's personal slide collection includes over 6,000 slides. For each slide, a reference number to it, a description of its botanical contents, and its relation to different parts of the course have been entered into a separate computerized data bank. Thus, when preparing a lecture on a given topic throughout the semester, Crovello merely has to request all slides with that particular subject number. Plant names are available in the description of each slide so that if Crovello wanted all slides of a given genus or of a given family he again merely has to input the particular names of interest. All such searches are relatively inexpensive, remaining below five dollars.

### Multiple Choice Course Review

Making multiple choice question sets available at remote computer terminals for use at the student's convenience is not new. In Plants and Human Affairs we use a rather simple driver program written in PL1 to present a question to the student, who then chooses an answer. After each choice the computer indicates whether his/her choice is correct or incorrect. But in addition, if the choice is incorrect, an explanation is given as to why it is incorrect. If the choice is correct, then additional subject information is provided. As teachers throughout the country have discovered, this extra information is a great incentive to students, even to the point of their deliberately choosing incorrect answers after they have chosen the correct one to obtain the explanation for the incorrect choices. The innovative feature of the multiple choice course reviews in Plants and Human Affairs is that the students themselves create the questions. Throughout the semester, as a certain section of the course is completed, students are required to submit a multiple choice question, five answers, and an explanatory reply for each of the answers. After being checked and accepted for correctness and suitability by a teaching assistant, the information is entered into the Plants and Human Affairs computerized data bank, and is accessible to all students. If certain questions, answers, or replies are incorrect or are not suitable, they are returned to the student who then can choose to correct and resubmit them. If the student brings the questions up to standards, he/she can receive almost full credit. Or the student can choose not to correct them and thus be satisfied with a lower grade than is possible. This option to enhance one's grade is a strong stimulus to most students.

Advantages of requiring students to create their own questions are several, including creating of the need for them to search their notes thoroughly and to go into the relevant literature to create a question, five answers, and five replies. Creation of the question is usually not difficult, but obtaining five answers and meaningful replies to these answers is. A second advantage is that this assignment has removed the need for short lecture quizzes, which though short, still take up valuable class time. In addition, students also learn that creation of meaningful multiple choice questions and answers is a difficult process. This realization regularly evokes a voiced, sympathetic feeling toward their teachers. Creating questions, answers, and replies also decreases students' feelings of being passive participants in the educational process, a continuous problem in large lecture courses. Finally, because they see their contributions integrated into a large multiple choice course review data bank from which they and their fellow students benefit, camaraderie among students increases.

Actual review of these large sets of questions in the computer data bank is *not* required. However, an incentive for students to do so is the inclusion of a certain number of actual questions from the data bank on examinations throughout the semester. This is motivation enough for the students to review the questions and to understand what they mean. Also important is that students' attitudes toward reviewing the questions change from the negative one of having to do another assignment to the positive one of taking advantage of useful study aid. A further advantage of ***not requiring*** students to review the multiple choice questions is the complete elimination for the teaching staff to check that each student has reviewed some required number of questions and answers.

Because many students do not have a background in computers and are unfamiliar with some aspects of activating data sets and programs from remote terminals, we have made access to the review of multiple choice questions as simple as possible. Students merely have to learn to log on to the system. Once in TSO's READY mode, a student simply types the code of the particular segment of the course for which questions are desired (for example, PAHA 1, PAHA 2). These data sets are available throughout the semester for review by the student at his or her leisure.

## Simulation of Plant Related Systems

Given the goal of the course, and of students' interests, emphasis on indepth studies of plant systems involves applied botany more than basic botanical areas. The two simulation models used in Plants and Human Affairs have three purposes: (1) to provide deeper insights and knowledge of the system, (2) to permit students to gain a stronger understanding of the complex interrelationships among parts of a system, and (3) to show that these relationships affect the ultimate behavior of the system. For example, what are the causes of the large rise in 1977 coffee prices around the world? Is it just due to the weather freeze in Brazil which killed 75 percent of coffee plants?

A further purpose of the simulation models is to bring about changes in behavior of the students that is based on more fact and understanding (for example, to be more of a vegetarian or not, or to change voting decisions). The first simulation model to be described is the endangered species model, and the second is a model of world agricultural production and consumption called the PAHA World Model.

## The Endangered Species Model

Written in the BASIC programming language for wider transportability, the endangered species model permits students to gain insights into what components of society and of the natural world affect the endangered status (actual or legal) of species. Simultaneously, students gain insights into how various other components of society interact and affect each other. Among the major components considered are the current endangered status of species, current status of endangered species legislation, the public's attitudes towards endangered species, economic status of the country, and environmental index. There are several ways in which the simulation can be played. Long before sitting down at a terminal, students are provided with detailed mimeographed information on the endangered species model, as well as the current status of endangered species, and of the other components which are available in the system (for example economic index, sociological index, etc.). Students are asked to do two things to help them to understand how the current status of endangered species came about: to determine what components interact directly with other components, and to determine what the values of different components have been on a relative scale over the last decade. For example, has endangered status increased, decreased, or fluctuated over the last ten years? Have the attitudes of politicians and of the public toward environmental protection and endangered species changed? Once students have made certain decisions about these two areas, then they input their information on relationships and relative values of the components over the last ten years into the computer model. Their printout gives them what the present status should be, based on the result of their assumptions and data. Then they know how close their ideas of cause and magnitude are from the truth. Of course there is more than one solution to this system and this in itself is a remarkable discovery for many students.

Another way to exercise the model is to ask students to depict relationships among the components and to provide what they feel have been the relative values of each component over each of the last ten years. The computer then will take these data and plot a forecast for the next ten years. Thus, students quickly can see how incorrect they were, and learn the consequences of our society remaining on the course on which it appears to be.

The endangered species model is a simulation, but we prefer to refer to it as qualitative simulation rather than a regular, more quantitative systems analysis. Given the problem of avoiding extreme complexity for students in a nonquantitative course, the problem of component definition, and the problem of obtaining reliable values of characteristics of each component (for example, the public's attitudes), it seemed best to keep the model in a more qualitative mode.

### The Plants and Human Affairs World Model

This is a relatively simple but realistic model of world agricultural production and consumption. Its source is the World III Model of the Club of Rome written in the Dynamo II computer language. Under the direction of Professors J. Forrester and Dennis Meadows at Massachusetts Institute of Technology, the World III Model has a subroutine called, Agricultural Sector With Exogenous Inputs. The Plants and Human Affairs World Model has been adapted from this. Our model simulates the interaction of such factors as land fertility, population, and total agricultural investment to produce values over time for food produced, land yield, and amount of arable land.

Students are given detailed mimeographed information on the Plants and Human Affairs World Model. Discussions on the model are held in class before the students sit down at a computer terminal. The students' goal is to study the behavior of the agricultural model given certain changes that they choose. Conversely, their assignment can be to determine what components of the model must be manipulated and what values these components must have to achieve a certain condition after some specific number of years. This is a rather more complex model than the endangered species model and it is presented after the endangered species model has been analyzed. After students have made their runs with results of various perturbations, the data are then integrated and analyzed to discover any particular patterns. A critique of the relationships among the components as well as of the particular form of equation relating them is also made. Again, the purpose of the simulations in the course is not to have students develop a more reliable or specific world model, but to gain insights into the complex interrelationships involved in an aspect of Plants and Human Affairs.

### Advantages and Disadvantages of Computers In Plants and Human Affairs

Much of the information reported in this section was obtained from an anonymous questionnaire which was distributed to all students at the end of the semester. In addition, constant reactions were received from students throughout the semester. We first make some general comments.

For many students, particularly nonscientists, this is the first experience with computers. All survive, and many find it to be less painful and more profitable than expected. We have sensed an increase in understanding of the course material, credit for which (although not rigorously tested) appears partly due to the use of computers. We found computers to be useful both to the professor in enhancing his teaching effectiveness, and to students in increasing their learning effectiveness. We have been able to teach better what we have long taught, and to teach easily what was previously difficult. We have seen increased interest by students in the subject matter, and have been able to permit students, at least in part, to learn at their own pace. By using the University-wide computer facilities, including computer terminals, we have been able to do this at little cost to the department. This is important, particularly in justifying computer use to colleagues in biology who feel that there are more important uses for limited departmental funds.

More specifically, the use of computerized information retrieval has saved large amounts of time in the location of slides for particular lectures and of literature references for specific segments of the course. This is especially true in Plants and Human Affairs because one reference or one slide often is relevant to several parts in the course.

In the future, we hope to permit students to use the literature data bank to help them in preparing term papers which must analyze in detail some aspect of Plants and Human Affairs.

Use of the computer for multiple choice course review has met with good success, particularly because there is no requirement for students to review any set number of questions on the computer. Many indicate that they have a much better grasp of the subject matter after reviewing the questions, answers, and replies. While at first some complain about having to create questions, answers, and replies, ultimately they feel that the forced experience of getting into the primary literature and of reviewing their notes is worthwhile. A current problem which we are seeking to rectify is the redundancy of several questions in the data bank. This requires monitoring by teaching assistants who in addition to being biologists also must be familiar with computerized file manipulation.

Our simulations of plant-related systems do not yet appear to have the best combination of printed matter, actual simulation, and class discussion. The biggest problem appears to be that the professor over-estimates the students' ability to grasp any kind of quantitative presentation (even in the form of a systems diagram) in the limited time available. Much more time appears to be required before students run actual simulations to permit them to understand more clearly what the purpose of each simulation is, and the mechanics of running and modifying these simulations. In addition, many students suggested on their questionnaires that more time be spent in class discussing the results and meaning of the simulations after they are run.

## Conclusions

Computers are used in an advanced undergraduate Plants and Human Affairs course which is open to all juniors and seniors at the University. Computers were introduced to enhance both teaching and learning in the course. Major uses involved information retrieval, multiple choice course review, and simulation of plant-related systems of relevance to society. Reaction to all aspects of computer use in general has been good, but efforts must still be made to enhance understanding and clarity of the more difficult assignments, particularly those involving simulation. We believe that students benefit from the use of computers in Plants and Human Affairs, and while computers will not replace good teachers, they can enhance their effectiveness.

## Note

Correspondence with regard to the contents of this paper should be directed to Professor Crovello.

## References

Crovello, T.J. 1974. Computers in Biological Teaching. Bioscience 24(1): 20-23.

# An Interactive Program
# for Carbon Labeling in Photosynthesis
# Written in ASCII PL1

**Roy E. Gereau**
*Department of Biological Sciences*
*Michigan Technological University*
*Houghton, Michigan 49931*
*(906) 487-2546*

## Introduction

A beginning student in biochemistry is often overwhelmed by the complexity of biochemical cycles. He is beset with many names and structures which initially appear very similar and are difficult to distinguish from each other, and the relationships between intermediates and products are often poorly understood.

In order to clarify the mechanics of biochemical processes, many instructors employ the technique of illustrating the introduction of a radioactively labeled carbon compound somewhere in a cycle, then following the labeled carbon atom(s) through the cycle. This method has the advantage of graphically demonstrating exactly "which carbon goes where," and helps to make relationships between structures easier to understand. Its inherent disadvantages are due to limitations of time. An instructor rarely has time to trace more than a few labeled compounds through a given cycle, and a discussion of the variation in possible labeling patterns from the same set of reactions is often too time-consuming to fit into the schedule of undergraduate courses.

The Calvin cycle of C-3 photosynthetic carbon fixation (Whittingham, 1974) seems to present more difficulties to students at both the undergraduate and the graduate levels than do most other biochemical cycles. This is largely due to the necessity of combining *three* reagents ($3CO_2$) with *three* cyclic precursors (3 ribulose 1,5-diphosphate) in order to produce a single recognizable end-product (glyceraldehyde 3-phosphate). This same complexity creates problems in the use of radioactive labeling as a teaching tool. In a first attempt to resolve some of the difficulties of both students and instructors in using carbon labeling techniques as a tool in understanding photosynthetic fixation of carbon dioxide, the author has developed the program CARBONFIX.

CARBONFIX is written in standard ASCII PL1 for use with the UNIVAC 1110 computer at Michigan Technological University. It incorporates no highly specialized features of the language, and should be relatively easily transferable to any installation which has the PL1 language and a core capaicty of 29 K or more.

## Program Use

A student is given a sheet of paper which lists the commands necessary to sign on at a demand terminal and to execute the program. He is expected to have had at least a minimal introduction to the cycle, and students are instructed to run the program with a written copy of the cycle (usually in the textbook) in front of them. From this point on, the program is self-explanatory, and the student needs no further outside instructions.

Output begins with a brief description of the first two reactions of the cycle. The program then prints out these two reactions, showing the complete chemical formulas and marking radioactively labeled carbon atoms with an asterisk (Figure 1). At this point, there are six glyceraldehyde 3-phosphates,

**Figure 1**

```
WELCOME TO THE WONDERFUL WORLD OF PHOTOSYNTHESIS.
THE FIRST REACTION YOU WILL SEE IS THE CARBOXYLATION OF UNLABELED
RIBULOSE 1,5-DIPHOSPHATE WITH RADIOACTIVELY LABELED CARBON DIOXIDE.
THE RESULTING 3-PHOSPHOGLYCERATE IS THEN REDUCED TO GLYCERALDEHYDE 3-PHOSPHATE.
RADIOACTIVE CARBONS ARE MARKED WITH A *.
HAVE FUN, AND KEEP TRACK OF THOSE G3PS!

                        H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)
                            I                 I                 I
                            C=O               C=O               C=O
                            I                 I                 I
          3 *CO(2)  +      HC-OH      +      HC-OH      +      HC-OH
                            I                 I                 I
                           HC-OH             HC-OH             HC-OH
                            I                 I                 I
                        H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)

                                     YIELDS

                   *COOH              COOH             *COOH
                    I                 I                 I
                   HC-OH      +      HC-OH      +      HC-OH
                    I                 I                 I
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)

                    COOH             *COOH              COOH
                    I                 I                 I
            +      HC-OH      +      HC-OH      +      HC-OH
                    I                 I                 I
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)

                          +  6ATP  +  6NADPH(2)  YIELDS

                  *HC=O              HC=O             *HC=O
                    I                 I                 I
                   HC-OH      +      HC-OH      +      HC-OH
                    I                 I                 I
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)

                   HC=O             *HC=O              HC=O
                    I                 I                 I
            +      HC-OH      +      HC-OH      +      HC-OH
                    I                 I                 I
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)
```

three of them unlabeled, and three of them labeled in the one-carbon position. Five of these glyceraldehyde 3-phosphates must be used to regenerate the precursors of the cycle, and any one of the six could theoretically be used in any one of five reactions. The user is therefore asked to make a series of decisions as to which glyceraldehyde 3-phosphate is used in each reaction where one of them is required. After each decision, the resulting reaction, and all reactions up to the next one requiring a decision, are printed out. Thus, the user is immediately able to see the results of his decisions in terms of the labeling patterns of cycle intermediates. (See Figure 2 for the results of the first three decisions.)

Figure 2

```
CHOOSE TWO OF ABOVE SIX GLYCERALDEHYDE 3-PHOSPHATES TO FORM
DIHYDROXYACETONE PHOSPHATE.  NUMBER G3PS IN ORDER OF PRINTING
AND ENTER TWO INTEGERS SEPARATED BY COMMA.
1,2
                              *H(2)C-OH          H(2)C-OH
                                   |                 |
                2G3P YIELDS        C=O       +       C=O
                                   |                 |
                               H(2)C-OPO(3)      H(2)C-OPO(3)

CHOOSE ONE OF ABOVE TWO DIHYDROXYACETONE PHOSPHATES TO REACT WITH GLYCERALDEHYDE
3-PHOSPHATE TO FORM FRUCTOSE 1,6-DIPHOSPHATE.
ENTER SINGLE INTEGER.
1

CHOOSE ONE OF REMAINING FOUR GLYCERALDEHYDE 3-PHOSPHATES TO REACT WITH
DIHYDROXYACETONE PHOSPHATE.
USE ORIGINAL SEQUENCE FOR NUMBERING; ENTER SINGLE INTEGER.
3
                                           H(2)C-OPO(3)           H(2)C-OH
                                               |                      |
                                               C=O                    C=O
    *HC=O          *H(2)C-OH                   |                      |
      |                 |                  *HO-CH                 *HO-CH
     HC-OH     +        C=O       YIELDS       |          YIELDS      |
      |                 |                    *HC-OH                 *HC-OH
  H(2)C-OPO(3)      H(2)C-OPO(3)               |                      |
                                              HC-OH                  HC-OH
                                               |                      |
                                           H(2)C-OPO(3)           H(2)C-OPO(3)
```

When all decisions necessary for the completion of the cycle have been made, the program prints out the final reactions. It then displays the labeled precursors (ribulose, 1,5-diphosphate) and the net photosynthetic gain in glyceraldehyde 3-phosphate produced by this turn of the cycle (Figure 3). The student then chooses whether or not he wishes to go through the cycle again. If he chooses to continue, he is given the choice of using the labeled precursors produced by the first turn of the cycle, or of starting over with unlabeled ribulose 1,5-diphosphate. If he chooses to stop, the program terminates execution.

Figure 3

```
RIBULOSE DIPHOSPHATE NOW AVAILABLE FOR CARBOXYLATION
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)
                    |                 |                 |
                    C=O               C=O              *C=O
                    |                 |                 |
                   HC-OH     +      *HC-OH     +      *HC-OH
                    |                 |                 |
                   HC-OH             HC-OH             HC-OH
                    |                 |                 |
                H(2)C-OPO(3)      H(2)C-OPO(3)      H(2)C-OPO(3)

NET PHOTOSYNTHETIC GAIN IN GLYCERALDEHYDE 3-PHOSPHATE:
                                     HC=O
                                      |
                                     HC-OH
                                      |
                                  H(2)C-OPO(3)
```

CARBONFIX can also be executed by means of punched cards from a card reader. The student is supplied with a list of commands needed to execute the program by cards, and with a list of all the questions asked by the program (Figure 4). He must decide upon his answers to the questions in advance, and submit them as data in his card deck. The program then executes as it would from a demand terminal, listing the student's answer immediately after each question. Use of the program from a card reader demands a little more understanding of the cycle by the user, but it provides flexibility of usage in situations where access to demand facilities is limited.

Figure 4

## Questions You Will Answer

Answer each question on a separate card, beginninng in Column 1:

1. Choose two of above six glyceraldehyde 3-phosphates to form dihydroxyacetone phosphate. Number G3PS in order of printing and enter two integers separated by comma.
2. Choose one of above two dihydroxyacetone phosphates to react with glyceraldehyde 3-phosphate to form fructose 1,6-diphosphate. Enter single integer.
3. Choose one of remaining four glyceraldehyde 3-phosphates to react with dihydroxyacetone phosphate. Use original sequence for numbering; enter single integer.
4. Choose one of remaining three glyceraldehyde 3-phosphates to react with fructose 6-phosphate to form erythrose 4-phosphate and xyluclose 5-phosphate. Enter single integer.
5. Choose one of two remaining glyceraldehyde 3-phosphates to react with sedoheptulose 7-phosphate to form ribose 5-phosphate and xylulose 5-phosphate. Enter single integer.
6. Do you wish to go through the cycle again? Enter yes or no. (If no, program terminates execution; if yes, keep going.)
7. Enter 'old' to reuse labeled ribulose 1,5-diphosphate.
   Enter 'new' to start over with unlabeled ribulose, 1,5-diphosphate.
8. Return to question 1.

### Program Structure and Limitations

Although it is reasonably complex in structure, CARBONFIX uses no esoteric or highly system-dependent features of PL1. Chemical formulas are initialized within a structure, BEFORE, in one- or two-dimensional character arrays. This structure remains unchanged throughout the program for ease of re-initialization, and all manipulation of radioactive labels takes place in a parallel structure, AFTER. The character manipulation features of PL1, especially the SUBSTR and INDEX functions, are used frequently throughout the program. The entire cycle is contained within a loop, CYCLE, re-execution of which depends upon the user-supplied value ('YES' or 'NO') of the character variable FLAG. Output of all chemical reactions is done by internal subprocedures. The program calls on no external subprocedures, system routines, files, or other external sources.

CARBONFIX is basically intended for use on a Texas Instruments, General Electric, or other remote terminal capable of producing a hard copy of eighty columns width. The speed with which reactions are printed and disappear from the screen on most CRT terminals would make the output of the program nearly meaningless to a person who was not already well acquainted with the cycle. In addition, it is often necessary to check past output for help in making choices among remaining glyceraldehyde 3-phosphates, and this would not be possible on a CRT terminal. Thus, at installations where access to hard-copy terminals is very limited or lacking, students would be virtually restricted to using the program from the card reader.

### Safeguards Against User Errors

Interactive programs are notoriously subject to sabotage by their users, and every effort has been made to safeguard this program against the typical types of errors. All numerical input is read in free format (LIST-directed), so that the position on the line or card of a user's response to a question is of no consequence. Character input (e.g., a response of YES or NO to a question) is accepted anywhere in the first twenty columns of a line or card by use of the INDEX function. In case of an inappropriate response to a question requiring character input (e.g., a user inadvertently types YS instead of YES), the user is informed of his error and allowed to answer the question again. Finally, by means of storing previous choices in a one-dimensional array, the user is prevented from "re-using" a molecule which has been used in a previous reaction. If he attempts to choose the same glyceraldehyde 3-phosphate twice, the program informs him of the reaction in which it was used, and directs him to choose another.

## Intentions and Results

A good deal of thought went into the construction of a program which would meet the needs of students learning the carbon fixation cycle for the first time. Naming each of the compounds in every reaction would have defeated the pedagogical purposes of the program by "spelling everything out," while leaving everything completely unidentified would have created a bewildering mass of symbols with little, if any, learning value. A balance was sought as follows:

The student is provided with a starting point by the complete identification of the reactants and products of the first two reactions (carboxylation and reduction) of the cycle. After this, the only compounds identified by name are the reactants and products of the reaction immediately following each of the user's decisions, and, at the end, the regenerated precursors and the end product. In this manner, the student is provided with a series of check points by means of which he can locate his present position in the cycle. Between these points, he must "follow through" from the named reactions, associating the names of the various reactants and products with their structures, and associating sequences of reactions with their position and function in the cycle.

Of course, this type of modeling of biochemical processes is by no means restricted to photosynthesis. The same and very similar programming techniques provide exciting potentials for the creation of teaching tools in other areas of biochemistry. An obvious extension of the present program would be to model the formation of glucose from the excess glyceraldehyde 3-phosphates of the carbon fixation cycle. A student could then follow the labeled carbon atoms of the glucose through anaerobic glycolysis and the tricarboxylic acid (Krebs') cycle. This program is currently in the planning stage.

As is usual with people who are not familiar with the use of a computer, some students seemed rather intimidated by the program on first introduction, but a single demonstration and a clear set of directions did much to alleviate this problem. Upon discovering that going through the cycle one time requires only fifteen to twenty minutes, many students were willing to try it at least once. Of those who did, more than one-half reported that they found it useful as a tool for learning both the components (names and structures of compounds) and the mechanics of the cycle. Several students commented that they used the output as a study guide while preparing for tests. In the spring term of 1977, a large biochemistry class will have access to CARBONFIX, and a brief summary [one or two sentences] of their reactions will conclude this paper.

## References

Whittingham, C. P., *The Mechanism of Photosynthesis*, American Elsevier Publishing Company, New York, 1974.

# Teaching Mendelian Genetics with the Computer

**James W. Small, Jr.**
*Department of Biology*
*Rollins College*
*Winter Park, Florida 32789*
*(305) 646-2129*

### Introduction

Most undergraduate biology programs teach the principles of inheritance at the freshman-sophomore level in an introductory course and at the junior-senior level in a "general genetics" course. Students in both courses seem to have great difficulty mastering the basic concepts of Mendelian Genetics and solving even simple problems. In three years of teaching the introductory course for science majors at Rollins College, never have more than 30 percent of the students enrolled been able to solve problems on examinations. However, more than 95 percent of these same students turned in correct answers to even more difficult "homework" problems. Although the percentage of students able to master test problems increases in the general genetics course, instructors of that course frequently complain about how poorly the students understand the concepts of Mendelian inheritance. We have tried different approaches to the subject in lecture, problem sessions, and additional "homework problems" without much success. As a new approach, the author decided to develop computer instructional programs for both our introductory and advanced genetics courses. This paper describes the programs developed and the results obtained from their use during the fall term of 1976.

The instructional programs developed had to satisfy several criteria. First, they had to present different practice problems to each student to prevent the obvious copying of correct solutions on homework. Secondly, the programs had to be elementary enough for the introductory student but able to be expanded to meet the needs of the advanced. Third, the programs had to allow the student to develop a method to solve the problem rather than give all information needed. The programs would therefore require the student to plan experiments and carry them out in order to gain the information to solve the problems, teaching not only the methods of genetic research but also the scientific method. Finally, the programs had to be realistic in that they have a counterpart in real life genetic research.

### The Programs

Program FLYSIM, a complex simulation model of the genetics of the fruit fly *Drosophila melanogaster*, is written in the BASIC-PLUS language for the PDP 11/40 time sharing computer at Rollins College. The genetics of this animal is relatively well known and mutant stocks are commonly used for both research and teaching. The initial version of the program allowed students to simulate the breeding of flies using either the wild or thirty-eight different mutant genes (Figure 1). These characters were apportioned among four linkage groups corresponding to the four chromosome pairs of the real animal. Up to three character pairs could be used in combination for any cross. The student had to specify a one, two, or three character cross and give the genotype of both male and female parent. Geotypes were

FIGURE 1. List of mutant genes and codes used in programs: FLYSIM, MORFLY and NUFLY. Each gene pair has a wild (+) and mutant choice representing two possibilities for a given chromosome location.

| Choice A | Code | Choice B | Code |
|---|---|---|---|
| RED EYE (+W) | 01 | WHITE EYE (W) | 02 |
| WILD CUT (+CT) | 03 | CUT WING (CT) | 04 |
| WILD FORK (+F) | 05 | FORK (F) | 06 |
| BAR EYE (B) | 07 | WILD BAR (+B) | 08 |
| WILD LOZENGE (+LZ) | 09 | LOZENGE (LZ) | 10 |
| WILD SCUTE (+SC) | 11 | SCUTE (SC) | 12 |
| CROSSVEINED (+CV) | 13 | CROSSVEINLESS (CV) | 14 |
| WILD VERMILI0N (+V) | 15 | VERMILI0N (V) | 16 |
| WILD MINATURE (+M) | 17 | MINATURE (M) | 18 |
| WILD SINGED (+SN) | 19 | SINGED (SN) | 20 |
| WILD SABLE (+S) | 21 | SABLE (S) | 22 |
| WILD ECHINUS (EC) | 23 | ECHINUS (EC) | 24 |
| WILD BROAD (+BR) | 25 | BROAD (BR) | 26 |
| CURLY (CY) | 27 | WILD CURLY (+CY) | 28 |
| WILD DUMPY (+DP) | 29 | DUMPY (DP) | 30 |
| WILD VESTIGIAL (+VG) | 31 | VESTIGIAL (VG) | 32 |
| LOBE (L) | 33 | WILD LOBE (+L) | 34 |
| WILD BLACK (+BLC) | 35 | BLACK (BLC) | 36 |
| WILD BROWN (+BW) | 37 | BROWN (BW) | 38 |
| BRISTLE (BL) | 39 | WILD BRISTLE (+BL) | 40 |
| WILD CINNABAR (+CN) | 41 | CINNABAR (CN) | 42 |
| WILD SCARLET (+ST) | 43 | SCARLET (ST) | 44 |
| WILD STRIPE (+SR) | 45 | STRIPE (SR) | 46 |
| WILD EBONY (+E) | 47 | EBONY (E) | 48 |
| WILD CLARET (+CA) | 49 | CLARET (CA) | 50 |
| DICHAETE (D) | 51 | WILD DICHAETE (+D) | 52 |
| WILD SEPIA (+SE) | 53 | SEPIA (SE) | 54 |
| WILD SPINELESS (+SS) | 55 | SPINELESS (SS) | 56 |
| STUBBLE (SB) | 57 | WILD STUBBLE (+SB) | 58 |
| LYRA (LY) | 59 | WILD LYRA (+LY) | 60 |
| WILD BENT (+BT) | 61 | BENT (BT) | 62 |
| WILD EYELESS (+EY) | 63 | EYELESS (EY) | 64 |
| WILD SHAVEN (+SVN) | 65 | SHAVEN (SVN) | 66 |
| WILD SPARKLING (+SPA) | 67 | SPARKLING (SPA) | 68 |
| STAR EYE (STR) | 69 | WILD STAR (+STR) | 70 |
| GULL WING (G) | 71 | WILD GULL WING (+G) | 72 |
| HAIRLESS (H) | 73 | WILD HAIRLESS (+H) | 74 |
| DELTA (DL) | 75 | WILD DELTA (+DL) | 76 |

specified by a pair of number codes corresponding to the genes of each chromosome pair. For a given character, the number codes input must be of the same set for both male and female. Figure 2 demonstrates the input of data for a two character cross of unlinked genes and the computer's output of offspring data. The phenotypes are specified by 1-4 character strings that correspond to the gene that is shown by the offspring. The student may backcross these offspring among themselves or to a new genotype. Not that the data follow genetic laws by showing the 9:3:3:1 phenotypic ratio typical of an unlinked dihybrid cross and that there is sampling error associated with each offspring number. A repeat of the cross would give a different total number. If the cross involved a sex-

FIGURE 2. Simulated computer display from program FLYSIM for a dihybrid cross. The genes black (BLC) and Bent (BT) are not linked as shown by the 9:3:3:1 phenotypic ratio.

```
FOR CROSS # 3

MONO-, DI-, OR TRIHYBRID CROSS?  GIVE 1,2, OR 3?  2

GIVE PAIR OF CODES FOR GENOTYPE OF FEMALE PARENT (E.G. 01,02).
FOR CHARACTER 1  CODES ARE?  35,36
FOR CHARACTER 2  CODES ARE?  61,62

GIVE PAIR OF CODES FOR MALE PARENT. IF SEX-LINKED, ONLY FIRST IS USED.
FOR CHARACTER 1  CODES ARE?  35,36
FOR CHARACTER 2  CODES ARE?  61,62


RESULTS OF CROSS BETWEEN:

          MALE                               FEMALE

    +BLC        BLC                  +BLC          BLC

    +BT         BT                   +BT           BT


    CODE              PHENOTYPE              NUMBER
     1                +BLC  +BT               4011
     2                 BLC   BT                437
     3                 BLC  +BT               1348
     4                +BLC   BT               1340


DO YOU DESIRE TO CROSS ANY OF THE ABOVE FLIES (YES/NO)?  N

NEXT CROSS
```

linked gene, the data would be output separately for both male and female. By examination of ratios and backcross data, the student could determine characteristics of the gene like dominance, lethality, linkage pattern, etc. Advanced students could use three character crosses of linked genes to both sequence and determine relative distances between genes (gene mapping).

A second program, "MORFLY," was developed from the FLYSIM program to teach students how to determine the genotype of parents from offspring data. MORFLY asked the student only to specify a one, two, or three character cross. The program then output offspring phenotype data and allowed the student to backcross offspring to offspring or to a new genotype. Through an appropriate test cross scheme, the student could determine the genotype of the parents. The genotype was then input to the computer which would tell the student if the answers were correct or not. If the answers were incorrect, the student was told to try again and the

FIGURE 3. Simulated computer display from program MORFLY or NUFLY. Note the backcross of the original offspring to a input genotype.

```
FOR CROSS #1


OFFSPRING OF UNKNOWN PARENTS:

        CODE                  PHENOTYPE              NUMBER

         1                    SN FEMALE               141
         2                    SN MALE                 139
         3                    +SN FEMALE              142
         4                    +SN MALE                138


DO YOU DESIRE TO CROSS ANY OF THE ABOVE FLIES (YES/NO)?  YES
ENTER '1' FOR CROSS OF ONE OF ABOVE PHENOTYPES TO NEW GENOTYPE
OR '2' FOR CROSS OF TWO OF ABOVE PHENOTYPES?  1
GIVE CODE FOR FIRST PHENOTYPE YOU WISH TO BACKCROSS?  3

GIVE PAIR OF CODES FOR MALE PARENT.  IF SEX-LINKED, ONLY FIRST IS USED.
FOR CHARACTER 1  CODES ARE?  20,20


RESULTS OF OFFSPRING CROSS

        CODE                   PHENOTYPE               NUMBER

         1                     +SN FEMALE                258
         2                     + SN MALE                 262
         3                      SN FEMALE                265
         4                      SN MALE                  263


DO YOU DESIRE TO CROSS ANY OF THE ABOVE FLIES (YES/NO)?  N
DO YOU DESIRE TO BREED FLIES FROM ORIGINAL CROSS?  Y
```

original offspring data repeated. After three incorrect attempts to solve the problem the correct solution was printed. Figure 3 shows a typical example of the solution of a MORFLY problem. A modified version of MORFLY called NUFLY recorded the student's name and access code in an answer file along with the correct answers to a one, two, and three character cross. The student handed his answers in to the instructor for checking against the answer file. This program was therefore used as a test of the mastery of test-cross skills.

The offspring data output by the program was realistic in that it followed genetic laws and had sampling variation associated with it. Dominant, recessive, lethal, unlinked, and linked genes were available. In linked character combinations, the percentage of offspring produced of crossover phenotypes varied with the distance between genes. In mapping studies, the results underestimated the linkage distance for widely spaced genes as in the real system. The program also incorporated various error-checking routines to prevent students from breeding individuals of the same sex, inputting nonsensical data, etc.

### General Algorithm of the Program

BASIC-PLUS is ideally suited to these types of programs due to its string handling capabilities. The data for each mutant gene were stored as a string in a single computer "word." The string included the linkage group (chromosome 1,2,3, or 4), the map location in "Morgan" units, whether the gene was normal or lethal, if it was dominant or recessive, and the phenotype string. The information about any particular gene was then plucked from the string as needed.

After student input of data, the genotype of both male and female was divided into all possible gamete combinations (egg and sperm). Gametes of both male and female were put together to form the offspring genotype. At this point all offspring have a probability of occurrence equal to one. The character combination was checked to see if the genes were linked. If the genes were linked and the genotype a recombinant, the distance between the two genes was computed and the probability of that gene combination occurring determined from an empirical equation developed from the data of Perkins (1962) on the relationship between gene distance and recombinant phenotypes. If three genes were linked, this probability was reduced by a factor dependent on the maximum distance between all three genes. This factor was developed from the coincidence coefficient-map distance relationship data of Weinstein (1958).

Each probability of occurrence was multiplied by the same random number of the range 1-1,500 to give the number of individuals of that offspring. This number was modified by a random percentage of the range -3 to 3 to simulate sampling variation. Each genotype was converted to a phenotype by checking for the presence of dominant genes. The number of individuals of each phenotype was computed by addition of offspring number associated with each genotype. This information was output to the student.

In a backcross, the student inputs the phenotype number code(s) and a search is done for a genotype corresponding to the input phenotype. The search is random since there may be many genotypes for a given phenotype. This procedure allows the student to examine all possibilities by using multiple crosses. The information is converted into number codes and the program repeats. MORFLY and NUFLY differ from FLYSIM only in the use of a series of random number generators and traps to determine the genotypes of the parents.

### Results

Students were introduced to the programs in both our general biology and genetics courses. They were given a set of crosses to perform using FLYSIM and told to analyze for linkage pattern, lethality, and dominance. The students were also told to practice with MORFLY in preparation for a test to be given with the NUFLY program. Students spent a great deal of time working on the programs and came to the instructors for assistance in greater numbers than ever before. On the written examination in general biology, 64 percent of the students were able to solve the dihybrid cross problem and 77 percent got at least partial credit. On test cross problems (MORFLY), 82 percent correctly answered a simple problem and 50 percent a more difficult one. These results were very satisfying and considerably better than the maximum of 30 percent in previous years. Although comparative data are not available for the genetics course, the majority of student comment indicates the programs were helpful. Student comment in general biology was overwhelming in favor of use of the programs despite the fact that they are very time consuming.

In conclusion, the computer programs seemed to have significantly improved student performance. Whether this improvement was due to the programs themselves or to the time they forced the student to spend on the subject is debatable and probably irrelevant. The experience has given the author many ideas on improvement of the programs as well as their presentation. It is hoped with improved versions and the right combination of lecture presentation and example, we can make even greater improvement in student performance in this important area of biology.

## Acknowledgments

The Alfred P. Sloan Foundation supported the development of the programs through a grant to Rollins College for computer-aided instruction. I would like to thank Dr. Kathryn L. Edwards for using the programs in her genetics course and for many suggestions on improvements.

## References

Perkins, D. D. 1962. Crossing over and interference in a multiply marked chromosome arm of *Neurospora*. Genetics, 47, 1253-1274.

Weinstein, A. 1958. The geometry and mechanics of crossing over. Cold Sp. Harb. Symp., 23, 177-196.

# Multimedia Approach to Business Data Processing Instruction

**Ronald D. Armstrong and Joel D. Stutz**
*Department of General Business*
*College of Business Administration*
*The University of Texas*
*Austin, Texas 78712*
*(512) 471-3322*

## Abstract

Approximately 2,500 undergraduate students at The University of Texas enroll each year in a course entitled "Introduction to Business Data Processing." Over one-third of these students are currently being taught in large (between 180 and 300 students) classes via a multimedia presentation. Every lecture is prepared as a four-screen slide show with the frequent use of supplemental aids, such as CRT's with video projection, 16mm film, and closed circuit television. The main intent of this paper is to describe an innovative approach to classroom instruction, along with the successes and failures of the system.

## Introduction

Since 1966 a three-hour course in business data processing has been required of all students on route to a BBA degree at The University of Texas at Austin. Approximately 2,500 undergraduate students enroll in this course each year. Until 1974 more than 80 percent of this number were taught by teaching assistants (TA's/student teachers) in classes averaging forty-five students. This high percentage of direct TA contribution created two major problems: (1) the accreditation rating of the College of Business Administration was low, and (2) the overall consistency of course content and quality was difficult to maintain. To alleviate these difficulties the college introduced large multimedia classes taught by full-time faculty members. In the fall 1976 semester, of the 1,132 undergraduate students enrolled in the introductory data processing course, 480 were taught in two multimedia sections and the remainder were taught in classes of approximately forty-five students each. Two of the smaller classes were also taught by full-time faculty. Thus, only about 50 percent of the students were taught by TAs.

Of course, large classes are of no true benefit unless the quality of education can be maintained and student prejudice against mass instruction can be overcome. To achieve this, over 1,200 man hours were spent developing multimedia presentations for the data processing course. We feel that this has resulted in an innovative approach to classroom instruction. Every lecture is given as a four-screen slide show with the frequent use of supplemental aids, such as CRT's with video projection, 16mm film and closed circuit television. The students can, while in the classroom, see: (1) a problem statement, (2) a step-by-step development of the flowchart to solve the problem, (3) the conversion of the flowchart to a computer language (in our case, BASIC), and (4) a computer execute the program with sample data.

Slides picturing computer hardware and data processing systems, in addition to title slides for concepts and definitions, assist the student in conceptualizing aspects of business data processing. Since the slides present the material in a clear, well organized, logical sequence, note-taking is enhanced. By having the complete lecture developed on slides, the instructor is not required to spend class time writing on the blackboard, and can maintain eye contact. Also, the large classes free TA's from platform teaching requirements so that they can conduct group-help sessions and give individual assistance at remote computer sites.

The next section will outline briefly the content of the introductory business data processing course. Section Three will describe the use of the multimedia facilities in the

28

classroom presentations. The successes and failures of these teaching techniques will be discussed in Section Four.

### Course Content

The primary objective of the course is to provide students with a fundamental knowledge of the interactions of computer hardware, software, and personnel in a business environment. In most respects the subject matter covered is similar to that found in other introductory data processing courses. Because it is felt that the student can neither appreciate nor understand the concepts of data processing without some programming knowledge, approximately one-half of the course is spent on flowcharting and learning the programming language BASIC.[3] BASIC was chosen because it is easy to learn and well suited to interactive terminal usage. Sample problems involving payrolls, inventory, interest calculations, and other business related topics are flowcharted and coded in BASIC. During the semester every student is required to flowchart, code in BASIC, and solve on the computer more than ten homework problems. Both batch processing and interactive facilities are utilized to do the assignments.

Three computer systems, remote batch facilities, and interactive terminals are available for student use. The main Computation Center maintains a Control Data Corporation (CDC) 6400/6600 linked in a dual environment as well as a Digital Equipment Corporation (DEC) DEC-10 system. The College of Business Administration has a stand-alone computer system as well as a remote batch station and interactive terminals for usage of the Computation Center facilities. The stand-alone system is a CDC 3100 used by introductory students for canned programs only. The remote batch station is a PDP 11/40 with low speed card reader and medium speed line printer for communication with the CDC 6400/6600. For interactive processing the college has a computer terminal laboratory with fifteen CRT terminals and twenty ASR Model 33 teletypes. Several terminals are hard-wired to the CDC 6400/6600, several are hard-wired to the DEC-10, and the remainder are on dial-up lines.

Computer hardware components which are described in the course include the Central Processing Unit (CPU) and its subunits, and devices for data preparation, input/output, and secondary storage. Only a very cursory description is given to the physical aspects of these components. The main emphasis is placed on the functional aspects. It is the intent of the course to stress what the devices do, when certain hardware should be used, and the relation to computer software.

Emphasis is placed on how the computer software utilizes the CPU. It is felt that it is particularly important, when discussing internal storage, to delineate the difference between an actual address, a symbolic address, and the contents of an address. These concepts are then carried over to the lectures on computer software. It is pointed out that the variable name is a symbolic address, the computer executes with the actual address, and the value of the variable is the contents of the address. This has proven to be most helpful when introducing the student to flowcharting and BASIC.

The textbook for the course is *Introduction to Contemporary Business Data Processing.*[2] This book was written at The University of Texas and is explicitly designed to describe the computer systems on the campus. It also contains all the material on hardware and software necessary for the course.

### Multimedia Presentation

Essentially the same material is covered in the regular-size classes with about 45 students per class, and in the 2 large classes with 180 and 300 students, respectively. However, the mode of presentation is entirely different. It should be recognized that a completely oral presentation would be very inappropriate for this type of class. Also, the use of the blackboard as a visual aid on a regular basis in the large classes is unrealistic for two reasons. First, unless the characters are written in extremely large print, students in the rear of the room can not read them, and second, a very limited amount of blackboard space is available in the assigned classrooms. Thus, the instructors in the large classes are forced to give some type of screen presentation. Because a four-screen slide show allows a great deal of flexibility and because the facilities for such a show are available at The University of Texas, this was chosen as the primary visual aid in the two large sections of an "Introduction to Business Data Processing." Rear screen projection is utilized in these facilities, permitting the projection of slides without the annoyance of light beams. The instructor is then free to walk about and point out items on the screens without a spotlight effect or casting a shadow.

Other visual aids which are utilized in the classroom are a 16mm film projector, overhead projector, television/video-cassette player, and a video-beam projector coupled to a CRT. The latter device allows the capability for realtime computer output to be displayed in the classroom.

Over 1,200 hours have been spent developing slides and scripts for the multimedia presentations. Thirty-five one-hour lectures requiring a total of 2,500 slides have been prepared. The majority of these slides are colored, and when several related topics are presented on a single screen, the topic being discussed is highlighted through the use of colors. Although the presentations over the past three semesters have been fairly stable, revisions and additions to the material are being made as technology changes or improved methods of exposition are recognized.

A frequently used technique in the multimedia presentations is to disperse the subject matter across the 7-foot by 7-foot screens. For example, when lecturing on the fundamentals of data organization (data hierarchy) the screen displays are organized as follows. Screen one contains words "Data Hierarchy" throughout the discussion. Screen two progressively discloses the different levels of data hierarchy; that is, (1) characters, (2) items, (3) records, (4) files, and (5) data bases. As a different level appears, the definition is displayed on screen three and then a series of examples are given on screen four. Thus, the student is able to see at one time the main topic, the relationship between data hierarchy levels, the definition of that data hierarchy level being discussed, and examples. The presentation techniques of progressive disclosure, item highlighting, and subject dispersion are effective pedagogical devices and are used often.

Another frequently used technique is to sequentially develop a flowchart and/or BASIC program on the screens. Handouts which give both the problem statement and complete flowchart/program are distributed at the end of class. This alleviates the majority of the note-taking burden and allows the students to participate actively in the problem solving process. When flowcharting or programming a problem, the problem statement remains on screen one for the students to reference during the lecture. After the problem statement is discussed, a system level flowchart or a solution technique is sequentially developed on screen two; that is, a series of slides, each appending an additional symbol to the flowchart, are presented and discussed. Next, variable names are defined on screen four. A program (detail) level flowchart is then progressively developed on screen three. If a BASIC program is desired, this can be done on screen two. The program may either be progressively developed with slides or it may be entered in an interactive mode via a CRT terminal. Through the use of a video-beam projector, the students can see everything displayed on the CRT screen. The program can then be run using sample data previously stored on magnetic disk. Thus, the students are able to see actual instructor-computer interaction while in the classroom.

The video-beam projector is also used to demonstrate the use of specific BASIC statements. Small BASIC programs, designed to illustrate the syntax and semantics of certain BASIC statements, are executed in class via interactive processing. This allows the instructor to display results which would be very difficult, if not impossible, to relate orally. Also, questions on rather obscure workings of the BASIC language can be answered in class by executing BASIC programs. Rather than guess what the BASIC compiler will do, the desired procedure can be executed and the results displayed. An additional benefit derived from utilizing a CRT in the classroom is student exposure to the computer system commands, BASIC editing features, and file manipulation procedures.

Other supporting devices utilized in the lectures include TV video-cassette player and 16mm film projector. The majority of the films and cassettes are supplied by computer vendors. They are used to augment the lectures and, in some cases, bring a little levity into the classroom. Examples of the former function include an IBM 16mm film entitled "Printing with Light" which is used in conjunction with the lecture on computer output devices, and a TV cassette (locally produced) on how to submit batch jobs at The University of Texas Business School. An example of the lighter side is an IBM film entitled "Coffee Break," which features the Muppets in humorous skits involving computers. A film distributed by McGraw-Hill entitled "Introduction to Flowcharting" actually combines humor with important flowcharting concepts. In this film a mechanical man is programmed to walk through a maze. A step-by-step flowchart of the man's movements is developed. Thus, with the aid of a little slapstick, students are introduced to a logical thought process involving repetitive operations (looping).

## Successes and Failures

The most apparent success of the multimedia presentations is the capability to teach more students effectively in a single section. This allows a much higher percentage of the total course enrollment to be taught by full-time faculty. Although many of the TA's are outstanding teachers, in general both the quality of the course and the consistency of the material covered has been improved as a result of the large multimedia classes. Because fewer TA's are now required for platform teaching duties, more individual assistance can be given to students outside of the classroom. TA's on three-quarter time (ten hours a week) are assigned to assist the faculty in providing outside-of-the-classroom help to students enrolled in the two large sections. TA's on seven-quarter time are assigned to monitor the activities of the business school's terminal laboratory. The latter group of TA's is available to all the students enrolled in "Introduction to Business Data Processing." While they are open to any questions concerning the course, the primary responsibilities of these TA's are to instruct students in the use of the available computer hardware and to assist in debugging BASIC programs.

The students have responded very favorably to the multimedia presentations. Both large sections always achieve maximum enrollment and the drop rate among the students in these two sections is considerably less than that of the smaller sections. Also, the University-wide course evaluations given at the end of each semester reflect student approval. Most students indicate that a multimedia presentation is a great improvement over a traditional blackboard presentation. They frequently give verbal compliments about the slide shows and "live" computer demonstrations. The students feel that the use of the CRT with video projection is a valuable aid for overcoming the "computer mystique."

The successes of the multimedia classes are documented in other areas. The majority of the students who elect to take the more advanced business data processing courses are those previously taught in the large sections of the required introductory course. Reports from instructors of other business school courses which utilize the material from the "Introduction to Business Data Processing" course verify that students from the large sections are more than competitive with students from the smaller sections. Certainly, the average student seems to benefit from the multimedia presentation.

The failures of the multimedia sections are similar to those found in all large sections. Some students do feel inhibited when placed in such a large group; thus, questions are not asked as spontaneously as in a smaller group.

However, the students who would ask questions regardless of the environment create a healthy amount of student-instructor interaction. It is, nevertheless, very difficult for the instructor to gain any degree of familiarity with the individual student. In our opinion this causes some of the students to feel alienation and, as a result, become apathetic about the course.

Finally, we would like to point out aspects of the multimedia presentations which indicate neither a success nor a failure but are merely a byproduct. A great deal more time is required to prepare a multimedia lecture. A script must be written and approximately 100 slides loaded for each one-hour lecture. The complete presentation must be carefully planned before class. Minor changes in the script from semester to semester require hours of work. Administering and grading examinations are also an order of magnitude greater in the large sections than in the smaller sections. Special rooms must be scheduled to accommodate the classes. Frequently students are asked to write flowcharts or BASIC programs on a test and it is a very time consuming task evaluating 300 responses to this type of question.

## Conclusions

The successes of the multimedia sections far outweigh their failures. A large number of students are taught by a single full-time faculty member while maintaining the quality of education. All the notes for the course are displayed on four 7-foot by 7-foot screens. In addition to the four slide projectors, a 16mm film projector, a TV videocassette player, and a CRT coupled to a video-beam projector are utilized. The students are able to see the subject matter clearly and sequentially developed in color with a variety of presentation strategies used for emphasis, retention, and interest. The material is reinforced and augmented with films, and a clearer understanding of programming is possible through the use of the computer in the classroom.

To the best of our knowledge, the "Introduction to Business Data Processing" course at The University of Texas utilizes multimedia presentations to a far greater extent than any other course of its type in the country. There would appear to be two main reasons for this. First, a great deal of time and money must be expended to develop the scripts and slides. Second, the equipment for producing and delivering multiple-screen, multimedia shows is not available at most institutions. Even at schools where this equipment is available, the multimedia facilities are generally ***not*** used in conjunction with computer related courses. We feel that the presentations discussed in this paper are rather unique.

The College of Business at The University of Texas is fortunate in having excellent media production and presentation facilities.[1] At universities where it is desirable to teach large sections in a computer related course, presentations similar to those discussed in this paper may be very beneficial and worth the investment. However, many of the teaching aids presented here are applicable to smaller classes as well. In particular, a CRT terminal with video projection is used for classroom demonstrations by many instructors in the College of Business. Also, many of the advantages of multimedia-augmented lectures can be realized through the use of single-screen, front projection. The equipment needed for small classroom applications should be well within the budget of most institutions.

## References

1. Balsley, R. and J. Stutz, "Multi-Media System Operation Manual," Graduate School of Business, The University of Texas, Austin, Texas, 1976.

2. Barr, R., J. Hultz and R. Shocket, *Introduction to Contemporary Business Data Processing*, Austin, Texas, Cybernetic Press, 1974.

3. Kemeny, J. and T. Kurtz, *BASIC*, Hanover, N.H., Dartmouth College Press, 1966.

4. "Planning and Producing Slide Programs," Eastman Kodak Company, Kodak Publication S-30, Rochester, N.Y. 1975.

# Use of a Computerized Forecasting Model in theTeaching of Basic Business Statistics[1]

**Cecil H. Meyers**
*Department of Economics*
*School of Business and Economics*
*University of Minnesota, Duluth*
*Duluth, Minnesota 58812*
*(218) 726-7283*

## The Problem

One of the problems encountered in teaching basic statistics to business or economics majors is the necessity of making a smooth transition from the simple one-variable time series extrapolation model of forecasting to the quite sophisticated step-wise multiple regression model which may use ten variables. The transition difficulty arises from a pedogogical inability to closely relate complex statistical measures to real world problems. In this writer's view, an explanation of this difficulty involves three interrelated facets of the problem. First, no totally satisfactory method exists to introduce all or most of the complexities of an operational environment in the classroom. Secondly, seldom does an opportunity exist to know how policy decisions based on the exercise would have turned out.

The most subtle facet and the third of the interlocking parts, but the one to which this paper specifically addresses itself, concerns classroom deficiencies. When complicated problems are to be solved, it is often impossible to complete the exercise in the classroom. A trip to the computer center may mean a delay of a day or more. Because of computer remoteness, the student is not always closely involved with all phases of problem solving. Even a teletype with quick access may mean some break in the continuity and timing of the problem's answer.

A solution to these three interrelated problems becomes possible by using a portable minicomputer with a fast thermal printer, data card input, and graphic output capabilities. This equipment can be in the classroom precisely at the time it is needed. Thus, given the forecasting model and its objectives as noted in this paper, and given immediate computational capability, many of the "delay in results" difficulties can be solved directly in the classroom. At the same time, some degree of realism is introduced into the classroom and we involve the student in a participatory exercise, the lessons of which should not be quickly forgotten.

The forecasting model described in this paper possesses the capability of easing transitional problems between simplicity and complexity, and between classroom artificiality and real world outcomes. There also is hope that when the model is completed, the student discovers a unity of technique and purpose which replaces the fragmented bits and pieces encountered when studying what may seem like isolated topics. If at the same time, the model can provide an interesting, novel experience, a bonus payoff occurs and the student's learning curve is elevated above the ordinary level. In addition, the student learns about assessing validity and reliability to and from the model's input and output. In the end the student has gained insights into the broad policy questions which emerge when analytical techniques are applied to real world data.

The main thrust of this paper deals with building a model involving time series data and forecasting analysis as related to the business and economic world. This paper describes the model itself, then discusses the basic assumption based on my experience. The author has tried at all times to keep the model building process related to classroom experience and to involve student, instructor, computer, and at least some facets of the so-called real world.

It should be added that the progression of discussion in this paper leads from forecasting a single time series variable to step-wise multiple regression analysis, but does not include the latter. In addition, the actual classroom presentation constantly involves questioning of data, techniques, relevance, applications of methodology, effectiveness, and efficiency. The wind-up emphasizes interpretation of results. Thus, analysis, policy questions, and "What would you do if. . .?" kinds of queries are foremost in the learning results expected from this experience.

## Background

It must be understood that the students addressed here are largely sophomores with a sprinkling of juniors. They have completed the beginning sequence of frequency distributions, central tendency measures, dispersion, probability, the binomial and the normal distributions, sampling, inference, and hypothesis testing. At this point, we are about to begin the analysis of time series data involving the identification and use of trends (regressions) as forecasting techniques. The students have had some computer programming and time sharing teletype experience. Clearly, the students have already reached a fair level of statistical and computational achievement.

The students have access to equipment ranging from the usual hand-held calculators to the CDC 6600 having passed through the Cyber 72 and 74 on the way. Some of this equipment is used for stepwise multiple regression. However, the equipment used with the present classroom scenario is, by some standards, relatively small and inexpensive. We are using Wang 2200 series equipped with a CRT, marked sense card reader, thermal printer, and a flatbed analog plotter. This equipment is mounted in a portable (wheeled) case, and can be readily moved from classroom to classroom. Programs are written in BASIC and are readily stored on and accessed through cassette tapes. The hardware was obtained through a matching grant of the National Science Foundation and has been of great value in the classroom.

## Description of the Model

The forecasting sequence begins with the usual dichotomy of the time series model, i.e., the assumption that a business time series may be decomposed into the usual four parts, thus:

$$Y_t = T_t \times S_t \times C_t \times R_t,$$

where: $Y_t$ = the value of the time series at time period $t$.
$T_t$ = the trend component expressed in the units of the problem, i.e., dollars, index numbers, etc.
$S_t$ = the seasonal component expressed as a percentage of trend.
$C_t$ = the cyclical component expressed as a percentage of trend.
$R_t$ = the random component expressed as a percentage of trend.

The equation simply says that the time series value $Y$ at some particular time period achieved the value $Y_t$ because of the influence of trend, seasonal, cyclical, and random forces acting on the variable.

In our classwork, we begin with the elements and caveats of trend fitting, i.e., straight line, parabolic and exponential, plus the moving average and free hand methods. Following the three or four class hours spent on developing these concepts—and given all the other background students have acquired—we begin to build the computer model described below. It should be pointed out that we do not write the program in computer language since the program is already "canned," but we carefully discuss the various procedures and note how the various statistical techniques are forged into a unified system leading to the final forecast. The computer model as it now stands, runs through the following sequence of steps:

1. Real world $Y$ data is fed into the computer using marked sense cards previously coded.[2] The program will accept up to ten years of monthly data. Among the unique features employed in using this model is that we feed in data which ends at least two years behind the current year. In other words, if the present year is 1977, we use data ending in 1974 so that at least two calendar years of actual historical data remain for assessing the reliability of the forecast.

2. The computer now runs a twelve-month centered moving average from which the seasonal index is computed and adjusted to total 1,200. The moving average calculations and the seasonals are hard copied on the thermal printer.

3. The program then seasonalizes the original $Y$ data and uses those values as the input to a least squares straight line trend calculation. As the program now stands, possible options such as choosing a parabola rather than a straight line are discussed but not employed in the program.[3] The trend ($Y_t$) values are then printed out along with the "a" and "b" values for future reference.

4. The trend equation $Y_t = a + bX$ is used to extrapolate the trend for the next twenty-four months. These results are then printed out for informational purposes and for making comparisons to the final forecast which is completed in the next step.

5. At this juncture, the program seasonalizes the extrapolated trend values. The resulting $Y_t$ values are the forecasted values. The students now have an opportunity to discuss, to assess risk factors in policy or decision-making, and to note that possible economic traps lie in wait for the unsuspecting and uninitiated decision-maker.

6. We now utilize the graphic capability of the computer configuration to provide a graph which shows: (1) the historical data as it occurred during the forecast period, and (2) the forecast in another color. Students are quickly able to see how it all turned out. This feature is significant and unique. The motivational effect on students is encouraging. There is enough difference between the methods employed here and the usual presentation to create additional interest.

7. It is obvious by this stage that an enormous amount of work has been done by the computer to sift out the trend, seasonals, and forecasts; prepare the graph; and present these results in a remarkably short period of time. We discuss the pros and cons of the extrapolation adjustment forecasting method, and hopefully, student insights do emerge. Students quickly perceive the need for carefully interpreting the mechanical forecast, and for deciding what reliance they would place on that forecast in making management or policy decisions. By simulating the world of practical affairs, the need soon becomes apparent for day-by-day interpretation and updating either by rerunning the program with new data or perhaps by using the Delphi method.

8. If we wish, we can now have the hard copy output pages quick-printed so that the total problem and solution become available to the student by the next day's class. Incidentally, the graphs are done in large size for classroom use. Subsequently, graphic materials can be run through a reproducing reduction machine and quick-printed for the next day's class.

9. The windup to the forecasting model occurs when the students calculate the error of forecast by comparing each forecasted month with the actual data for the month. This could be accomplished by the computer, but is currently done manually since it can be done quite quickly with good impact on the students' retention mechanism. In fact, the errors are calculated, squared, and summed by quarter and year. Forecast reliability can thus be assessed.

## Assumptions of the Model

In order to develop the forecasting model, it was necessary that students understand certain assumptions underlying the procedure and the structure. These assumptions are discussed in class as we construct the model:

1. The data fed into the model are time series data, and it is assumed that the data are of value to the forecasting requirements of the problem. In other words, we are consciously assuming that the future is somehow reflected in the input information. In some respects, we have made a heroic assumption since the future is never identical to the past, and may actually be quite different.

What this particular assumption of repetition does, however, is make the student aware of the beauty and utility of using statistical means to sift out and identify patterns which occurred in the past and likely will be recycled in the future. Identification of the pattern, duration, amplitude, and intensity must be evaluated as to usefulness in forecasting. Incidentally, at this stage in the development of the model, the student begins to perceive the value and the limitations of experience as a guide to the future because of the need to question what data to use, how much data to use as input, and how much data to take away as usable output. Later, when the class moves to regression, the experience gained in dealing with these questions is easily transferred to regression problems.

2. As the time series data are conceptualized here, all input values are equally weighed. The student may wish to question this assumption, and the insight gained in doing so is important.

3. It is now necessary to choose a number of years of data experience to input to the problem. The period selected is, in effect, a time capsule of historical experience. The question is: Should valuable historical experience containing important keys to the future be assumed to be embedded in a capsule of five years of data? Six? Eight? Ten? More? It is valuable for the student to mull over these choices, again gaining insights into the deeper nature of the philosophical and theoretical questions involved.

In the actual running of the model, we are limited to using up to ten years of monthly data—because of the limited memory (16K) of our portable equipment. Students may choose, realistically, between five and ten years. It is useful for the student to see how much of the past can be discounted (if any) and at what rate. This, of course, also links with the problem of equal weights for each bit of data.

4. It is here assumed that the most significant repetitive patterns are the trend and the seasonal. It is also assumed that these patterns are identifiable by the usual time series methodology. Once identified from the historical data, it is assumed the patterns can then be applied to forecasting the future. The business cycle and random factors are discussed in terms of how knowledge of these influences can be identified and applied to the model. If so, how could the cyclical and random information be used in policy decisions? Moreover, we may ask, is it possible to bring in the random and cyclical factors by judgment and then use that judgment to refine the statistical forecast once the mechanical segment of the problem is completed?

5. It is assumed that the data series employed for forecasting purposes (industrial production, GNP, personal income, employment, etc.) is reasonably accurate as reported and is valid, i.e., measures what it says it measures.

## Contribution of the Model to Learning

As we build the model over several class periods, what kinds of intellectual residue may be expected as output from this learning effort?

1. The student learns to apply statistical tools in a logical manner, bringing them to bear on the solution of a realistic problem. Much of what was previously covered in the course can now be applied. From the discussions, the student knows that there is no single solution to a management-type problem, and that reasonable well-informed persons can disagree (within limits) on the proper problem-solving approach and the ultimate interpretation of the final results.

2. The student gains insights into the use and the limitations of the computer as applied to a problem which would be very time consuming if done by pencil pushing, i.e., more primitive methods. In fact, the student

perceives that forecasts of this somewhat involved type would probably be eschewed if it were not for the availability of the computer. Thus, in the absence of the computer, judgment would reign supreme. In addition, the student gains some classroom experience in actually seeing the mini-computer run and encountering the operational problems which crop up from time to time.

3. The student senses the necessity to be circumspect about the reliability, validity, and accuracy of the data used to mirror the historical experience of the variable. How were the data collected? What chances of error may lurk in the collection and reporting process? If the data contain errors, what about the reliability of the forecasts?

4. The learning developed here provides a vehicle through which a student can participate in the logic and development of the model. In other courses further down the academic line, students may build their own models to achieve particular purposes. At least here the students' eyes are opened to possibilities and alternatives available in future problem situations.

5. Running the model provides an opportunity for the student to interpret output in terms of the "real world." We can do this because we forecast for periods of time already completed. For example, we forecast 1974 and 1975 from historical data running through 1973. Or, we can go back even farther, and forecast for 1971 and 1972 from data running through 1970, etc.

In the author's opinion, this feature of seeing how the model "turned out" is one of the most valuable experiences arising from this exercise. The student's risk-taking proclivities can be tested by asking how much reliance he or she would place on the forecasts made for up to two years. Interpretation of results thus becomes a personal, meaningful experience for the student.

6. Student motivation is enhanced by getting involved in the performance at all stages of the experiment. Not every student is carried to new and enduring heights of intellectual attainment, but most show a level of interest above the usual classroom situation.

7. Use of the flatbed plotter is a unique feature of the total experience which enables us to show the forecasts plotted against the actual data as they turned out. This immediacy feature provides impressive visual demonstrations of the reliability of the forecast.

8. Once the graphic output is shown to the student, it becomes clear that total reliance on a mechanical model—with the assumption that the past is repeating itself—is not a completely reliable way to forecast. The student begins to note the need for human judgment based on experience with the forecasted variable in the melee of the real world environment. Judgment thus becomes an essential part of practical forecasting. Thus, interpretation of the mechanical result leads to the necessity of extending the model into nonmechanical avenues. The risk of applying mechanical output to the world in the absence of judgment then leads to a discussion of the Delphi method. This method becomes all the more relevant when the next point of atypicality is considered.

9. In terms of the original data as well as the forecast errors which become strikingly visible in the graph, the student may well perceive that atypical situations do happen. The effects of an inflationary bout, a recession, a period of drouth, or simply a run of bad luck (shades of Murphy's law) are perceived as altering the course of economic history—at least as far as the forecasted variable is concerned. Then the question emerges as to what can be done about interpreting atypicality. To answer this question the student must go well beyond his statistical knowledge and bring to bear economic theory and business course material.

It is the author's opinion that in studying mathematics, economics, and other courses, the student slowly and imperceptably comes to the impression that all data are well behaved, that the world is linear, or parabolic, or exponential, and a function can be found to turn out the precise value of the variable whenever "X" is plugged in and the crank is turned. What is not well behaved in the world may be easily assumed away. However, one must be careful of such assumptions when looking at the actual world of nature. It is a valuable experience for students to see the theories they know are tools—a set of keys for understanding—and that they are faced with the use of their intellectual equipment to make a judgment about the data and the forecast.

It is a matter of surmising at this point, but it appears that students begin to perceive their future role as possible managers of resources and come to expect to be called upon to provide creative solutions to problems they must face. Consequently, students can ask questions such as: (1) Why are certain events atypical? (2) How would I react to a management situation involving atypicality in terms of inventory policy, employment policy, marketing, etc.? (3) What are the risks of a "bummer" forecast?

At this point, an introduction to the possible use of regression as a forecasting tool can be explored. Would the forecasts be improved if additional variables could be brought to bear on the problem? If so, what variables might be selected as trial cases? Suppose these variables were run in an experimental test case, how would one judge the contributions of the new variables and how would the regression information be used in conjunction with the present model? Or, should the present model be thought of as an interesting exercise leading to the real challenge which is multiple regression?

### Comment

There are a number of spin-offs from this exercise. In varying degrees, students see the need for using their own intuition, understanding, and resources in problem-solving and interpretation of data. They see the necessity of dealing with the future in a rational, systematic manner. They see the need to formulate the problem, to evaluate the worth of the problem, and to see the problem in terms of "one among many" pressing issues requiring attention in the work-a-day world. Also, the student understands the use of statistical techniques applied to wringing the data for all they are worth, and the requirement of efficiency in the use of time, equipment, output verbiage, and the need to take risks in deciding issues on the basis of incomplete information. Obviously, there is good reason to believe that the experience reported in this paper is valuable and well worth the resources used.

### Student Reaction

How do the students feel about spending a week developing a forecasting model and then observing how well the forecast fits the facts?

The reactions we have up to this point are encouraging. Students report that for the first time they encountered a problem which seemed important, but lacked pat answers to neatly phrased questions. Over a year ago, one student, the son of a prosperous business tycoon, reported that he had held long discussions with his father on the model used, and that together they were going to apply the ideas to the business.

Several classes of students have seen the model in its developing stages, but only one class has (at the time of this writing) seen the fully operational model. The preceding classes have evaluated this "experiment" very highly, with about 80 percent exceptionally favorable response. The class which has seen the model in its present fully operational status was a late-afternoon class of twenty students, and the favorable score here was seventeen out of twenty. The author takes some comfort from the preliminary results, but he has been in the classroom long enough to know that no technique is capable of awakening every student to the beauty and relevance of the material in the course. Nevertheless, we feel that this model, which uses the minicomputer with the sense of dramatic technology which these machines produce, has something to do with the general acceptance of the total procedure. In addition, the students are sophomores (with a sprinkling of juniors). They have completed the basic economics course and are usually taking accounting while they are enrolled in the statistics course. These students are majors;

they are committed to a school of business program, and for the most part are well motivated.

## The Future

As presently conceived, the author believes the model fits a particular classroom need. There is no room for smugness, however, and there is always reason to try new approaches and to modify existing ones. In this case, a number of options could be added to the model. For example, the *Y* data could be plotted before any trend analysis is attempted. Then students could examine the graph and decide what type of trend to employ for the forecast. The graphic presentation can also be studied for atypical values which could bias the forecast. Also, several trend-types might actually be run through the computer and fitted to the data using the sum of squared errors $e^2 = (Y_t - T_t)^2$ to aid in the decision as to which approach should be utilized in the final forecast.

Other options exist. The model can subsequently be related to forecasting problems associated with the weighted influence of other independent variables as revealed by a step-wise regression exercise. Further work needs to be done on the possiblities existing in a marriage of trends and regressions.

Finally, one must resist the temptation to become too complicated at this point and turn the approach into a genuine forecasting course, which would divert us from the original objective which was to establish a forecasting perspective for students enrolled in the basic statistics course.

## Notes

1. The computer forecasting program segment of the model was supported by a grant from the University of Minnesota Council on Liberal Education Small Grant Program.

2. For example, the *Y* data could be industrial production, personal income, currency in circulation, money supply, installment credit extended, average hourly earnings, employment, etc.

3. These options may be introduced later.

# The Lease/Purchase Decision

**H. Gary Kuhlmann**
*California State College, Dominguez Hills*
*Dominguez Hills, California 90747*
*(213) 532-4300 Ext. 486 or 535*

## Abstract

Many classroom computer applications consist of games such as "Star Trek" or "Tic-Tac-Toe." While these games are entertaining, their educational value and practical application are limited.

The "Lease-Purchase Decision" is a computer program with a pragmatic application, since most students are faced with the problem of securing housing when they enter college.

Of course, the vast majority elect to rent a house or apartment. However, with inflation driving rents up, many students are wondering if they would be better off financially to purchase a small house for their four-year college career. Graduate students with children are especially sensitive to this issue. With the aid of their parents, loans, or part-time jobs, many students are financially able to purchase a house. The reasons that many of them do not are that:

1. They have never considered the purchase of a home as a viable alternative to renting.
2. They do not understand the financial terminology and legal aspects involved.
3. They are unaware of the advantages of home ownership in an inflationary society, or the advantages of leasing for short periods.

This Lease-Purchase Decision helps make college students and others aware of the differences between leasing and purchasing a home. The program is designed to analyze the data for each house, including price or rent, downpayment, taxes, utilities, and other costs of leasing or purchasing. In addition, the computer printout is personalized with the income of the student, number of dependents, and the individual's tax advantages based on a lease or purchase decisions.

## Introduction

The Dominguez Hills campus of California State College is a small school with a large percentage of minority students. Faculty and students have been developing and using a computer program which will aid minorities and other students in making housing decisions. The program is designed to determine for an individual whether he will be better off financially renting or owning a house during his relatively short term as a student-consumer.

The computer program is designed for college students and consumers who are aware that the purchase of a home will be an asset to them, both during and after college. For students who will not be permanent residents in the campus community, the purchase of a home can be risky since the purchase, closing, and eventual selling costs may be higher than the costs of renting for a short term.

This computer program has been designed to take into account the total costs of home ownership, including the downpayment, costs of money, length of the mortgage, costs of repairs, taxes, etc. The program also evaluates the student's income, number of dependents and tax position; as well as the length of time he/she will own (or rent) the home. After analyzing this data, the computer calculates the total cost of renting or purchasing a specific home each year for as many years as the student wishes, and tells him whether he will be better off buying or leasing.

The output is printed for each year, because appreciation (the increase in the value of a home) usually makes a home purchase a better investment each year it is held. However, if a student must sell or move after college or in a relatively short term, the computer output likely will tell him that it will be financially more advantageous for him to lease a dwelling, and thus avoid the costs of purchase, sales commissions, and other fees associated with home ownership.

### General Information and Assumptions

In the Lease-Purchase Decision program, a housing decision is made based on cost. The personal preferences of the individual (i.e., whether he or she is oriented toward suburban home ownership or apartment living) are not evaluated. It is assumed that the individual can afford to incur the costs of owning. In other words, cost is a determining factor, not a precluding one.

The dollar amount that an individual can afford to spend for rent and be as financially sound as if he/she had purchased a house depends on several factors: (1) The purchase price, downpayment and other terms of purchase, (2) the dollar returns that could be earned on alternative investments, (3) monthly amounts to acquire and maintain a home, (4) tax savings as a homeowner, and (5) net sales proceeds from selling the home at the end of the ownership period.

### Program Inputs

An individual using the Lease-Purchase Decision must input eighteen required variables. These data are used to calculate the "balance available for monthly lease." This figure represents the amount a student could spend for rent and be as well off financially as if she/he purchased a home. The lease or purchase decision hinges on this figure. If a comparable rental unit, including utilities, is available for not more than the amount calculated, the individual should lease the house. If, on the other hand, a comparable rental unit can only be obtained at a higher cost, the individual would be better off buying.

The computer program makes this calculation and prints the result for the end of each year of the holding period. The user can thereby compare the relative costs and advantages of purchasing or leasing for each year, and make an intelligent decision based on his annual financial advantage.

The student can input an optional twentieth variable, "Comparable Shelter Monthly Rent Including Utilities." "Purchase" is then printed if "comparable rent" is greater than or equal to "balance available for monthly lease." Conversely, "lease" is printed if "comparable rent" is less than "balance available for monthly lease."

Another optional variable, "non-real estate deductions," can be entered if the individual knows his/her deductions for charitable contributions, medical expenses, moving, non-mortgage interest, and other non-real estate itemized deductions. This optional specific input is then used for tax calculations instead of the programmed estimate of non-real estate itemized deductions.

### Program Operation

In order to participate in the program, the student enters the community and selects a house that is "for sale." Most of the information required for the program is available from the real estate broker who offers the property for sale. This includes:

1. Puchase price.
2. Down payment.
3. Mortgage interest rate.
4. Mortgage term (length of loan in years).
5. Mortgage insurance premium, monthly amount.
6. Settlement costs paid by buyer.
7. Monthly real estate taxes.
8. Monthly property insurance.
9. Monthly maintenance and repairs.
10. Monthly utilities excluding telephone.
11. Other monthly shelter expenses.

Personal data—information concerning the prospective tenant/purchaser, and his/her plans for the property—are then determined. These include: (1) Annual income, (2) number of dependents, (3) projection period (years planning to own house), (4) annual appreciation rate, (5) selling commission, (6) other selling expenses, (7) interest rate earned on purchase outlay if invested, (8) optional—

non-real estate deductions, and (9) optional—comparable monthly rent.

Any of these data may be changed, thereby providing additional program flexibility and utility. For example, the student can change the downpayment or the term of the loan, or any of the other twenty variables. The computer will calculate the annual result and its effect on his decision to lease or purchase. From this annualized output, a student (or any other consumer) can make make an informed decision to lease or purchase a home based on dollars and sense, not simply desires and wishes.

This program was tested with several classes at California State College, Dominguez Hills. The results, both positive and negative, were carefully documented and recorded.

### *Positive Results*

1. Many students, especially non-business majors, were amazed and pleased that significant income tax savings were available to home purchasers.
2. Several students were surprised to learn that the total cost of owning a home, after tax considerations, was more reasonable than they had expected. Others were surprised at the down payment and initial costs, as well as selling costs involved.
3. Many students, who are already homeowners, were pleased with their past decision to purchase.
4. Some students who were about to graduate or leave the campus community for other reasons were happy to see that renting made more economic sense for them in their short-run outlook.

### *Negative Results*

1. Some students, especially non-business students, were not able to understand all parts of their computer output.
2. Some students requested bi-lingual help in interpreting the input data required and the output information provided.
3. Many students felt that a users manual or information pamphlet should be made available prior to the distribution of the data request blanks. This users manual would explain the concept, purpose, and scope of the program, as well as the uses of the expected results.
4. Members of the relatively large Chicano population in Southern California felt that the users manual and the computer output itself should be made more applicable to them.

### *Suggestions for Improvement*

1. Develop and print an instruction booklet or users guide with information on input variables and explanations of the meaning of all output information.
2. Translate the instruction booklet or users guide into Spanish or any other language used by a substantial proportion of the students (or their parents).
3. Develop a new format for the computer program output with explanations printed directly on the output. This would be especially helpful if the users guide were unavailable or lost, and the computer output was the only material available to a student, consumer or parent.
4. Translate the output directly into Spanish or any other language, and format it to print directly from the computer.

### *Improvements Initiated*

These suggestions should be implemented by the time the papers are presented in June 1977. At this writing, three of the suggestions are being effected. Research has been initiated to: (1) Develop a comprehensive users manual, (2) translate the users manual into Spanish, and pre-test it, and (3) develop an output format with an explanation of each phase of the output printed by the computer directly on the output.

# Integrating Computer Concepts Into Principles of Accounting

**Henry J. Beck and Roy James Parrish, Jr.**[1]
*Danville Community College*
*Danville, Virginia 24541*

## Introduction

The use of the computer as a tool of business has made the integration of computer concepts and the integration of computer problem-solving capability a necessity in the principles of accounting course. The widespread use of credit by individuals and business has increased the number of business transactions which must be recorded, classified, and summarized. In addition, various governmental regulations have required business organizations to provide accurate accounting information in great volume and at tremendous speed. Business has met this increased demand for accounting records and information through the use of the computer. In fact, there are few areas of management that do not require at least a cursory knowledge of automated recordkeeping procedures.

The traditional approach to the teaching of accounting requires the student to demonstrate mastery of accounting concepts through the manual preparation of journals, ledgers, and financial statements. The tedious and time-consuming mathematical computations and manipulations required to record, post, and summarize accounting data often discourage students from further study in accounting. The detail calculations are stressed more than the accounting concepts. The presentation of these traditional manual methods to the accounting student of today gives the student an unrealistic view of the accounting profession. The accounting student of today must be permitted extensive exposure to the use of the computer as a valuable tool of the accountant. This exposure should be acquired through the integration of computerized techniques in the principles of accounting course.

## Problems of Integration

The integration of computer concepts into the principles of accounting course (as well as other subject areas) has presented significant problems in the past and continues to do so. One of the most significant problems relates to the degree of technical knowledge concerning computers required of the accounting student. This problem restated is: "Should emphasis be placed on writing programs or should emphasis be placed on the use of the computer as a tool for solving accounting problems?" The authors believe that emphasis is best placed on the use of the computer as a problem-solving tool, particularly in the principles of accounting course. In actual practice, the accountant probably will not be required to write programs. Rather, he will be required to use software systems or programs developed by specialists in computer programming. The accountant is most concerned with the capabilities of a computing system and needs to have a familiarity with input preparation and output interpretation. These statements are not intended to infer that the accountant should not be able to program a computer; however, a knowledge of procedures and computer problem-solving capability is more valuable than programming skill.

The source of the instruction concerning computer concepts and procedures is an area of concern. Should the accounting student receive this instruction in courses offered by the data processing department or in his principles of accounting course? The data processing instructor in most cases is not an accountant and may tend to emphasize applications of data processing other than accounting. The accounting instructor may lack the technical expertise needed to meaningfully integrate computerized procedures into the principles of accounting course.

Additional problems are created by attempts at integration because the accounting student is required to gain proficiency in two very specialized disciplines—accounting and computerized techniques. Skill must be acquired in the operation of new machines (i.e., the keypunch). In addition, new terminology must be incorporated into the student's vocabulary (i.e., card, record, edit, printout, etc.). During this time, the mastery of accounting principles must be occurring also.

Probably overshadowing all other problems of integrating has been the lack of satisfactory instructional materials. Until recently, instructional materials in this area applied only to small segments of the accounting cycle or were considered games to be played by students.

### Development of an Instructional Package of Computer Programs

In the summer of 1973, members of the accounting department and data processing department of Danville Community College began a joint effort to develop instructional materials that could be used to integrate computer concepts and automated procedures into the principles of accounting course. The development of these materials began with these assumptions:

1. The principles of accounting student does not need to be able to write computer programs.
2. Computerized accounting concepts should be presented in the principles of accounting course.
3. Materials developed should reflect current accounting practice and be compatible with most principles of accounting textbooks.
4. Materials developed should illustrate the use of the computer in all phases of the accounting cycle (recording, classifying, posting, summarizing, and statement preparation).
5. The use of the materials (computer programs) should require the student to demonstrate a knowledge of accounting concepts.
6. Data entry and detection of keypunching errors would be made as simple as possible for the student.

With these assumptions in mind, a package of instructional materials was developed at Danville Community College. This package consisted of coding forms for recording accounting transactions, a series of accounting problems, and a group of computer programs used to produce a set of accounting documents.

The programs consisted of two groups of programs referred to as Phase I and Phase II. The programs of Phase I accomplished the recording, classifying, posting, and summarizing functions of the accounting cycle. Phase I produced a sales journal, purchases journal, cash receipts journal, cash payments journal, general journal, general ledger, and trial balance. Phase II performed the adjustment and statement preparation function. Adjusting entries were recorded in the general journal, posted to the general ledger, and an adjusted trial balance, income statement, capital statement, and balance sheet were prepared by the Phase II programs. Student input cards were also checked for keypunching errors in each phase with an appropriate error message being produced for each keypunching error detected. Journal and statement totals were checked at key points in Phase I and Phase II to determine if they were in balance. The detection of a keypunch error or totals which were not in balance caused the programs to terminate.

The programs were designed for flexibility and were capable of processing transactions relating to sole proprietorships, partnerships, or corporations. The capability to establish beginning account balances was also provided.

### Solving Accounting Problems Using a Computerized Approach

To use the package of computer programs and related materials developed at Danville Community College, the student must perform all analysis that would normally be performed using manual methods. Figure 1 is a typical accounting problem that might be encountered in a principles of accounting course and illustrates the use of Phase I. The student would read each transaction and decide which journal should be used to record the transaction. Coding sheets are provided for coding entries to be recorded in each journal (Figure 3—sales journal, purchases journal, cash receipts journal, cash payments journal, and general journal coding forms). The student then records the transaction on the proper coding form. Such information as the date of the transaction, the account number of the account to be debited or credited, a transaction identifier (check number, invoice number, etc.), name of customer, amount of the transaction, discount basis or sales tax, discount percentage, and journal code. The account number is selected from the chart of accounts (Figure 2) provided for the student. The student must perform the above operations for each accounting transaction.

When all transactions have been recorded on coding sheets, the student punches each line recorded on the coding sheet into a separate card. When punching is completed, the deck is assembled by placing two identification cards at the beginning of the deck. If the beginning account balances are zero, a "+" card is placed after the identification cards followed by the transaction cards. An "*" card is placed at the end of the deck. The job is then processed by the computer. If no keypunching errors are detected, the sales journal, purchases journal, cash receipts journal, cash payments journal, general journal, ledger, and trial balance are produced (Figures 4 through 10). The package has the capability of producing selected journals. If keypunching errors are detected, the student corrects the error or errors and resubmits the job.

Figure 11 is a continuation of the problem shown in Figure 1 and illustrates Phase II of the package of computer programs. The student again analyzes the transactions and records the adjusting entries on general journal coding forms. Credit amounts are preceded by a minus (-) sign. Student decks are assembled as described in Phase I. Beginning account balances for Phase II can be keypunched directly from the trial balance produced by Phase I. The account number is punched in cc 1-3 with the amount punched anywhere in cc 4-80. The deck is then submitted for processing. If keypunching errors are not detected, the adjusting entries are recorded in the general journal, posted to the ledger, summarized in the adjusted trial balance and financial statements are prepared. These documents are shown in Figure 11 through Figure 16.

### Incorporating Computerized Techniques into the Principles of Accounting Course

The instructional accounting materials utilizing the computer developed at Danville Community College are not used to replace the traditional accounting textbook. The role of these materials is to reinforce those accounting concepts presented in a traditional accounting text. The student is introduced to accounting concepts using the traditional manual approach to the solving of accounting problems. After this introduction, the computer is used to reinforce the concept. The computer, of course, relieves the student from performing most tedious mathematical computations. Since the programs developed are applicable to problems relating to sole proprietorships, partnerships, and corporations, they may be used throughout the principles of accounting course.

The programs allow the instructor to present a more realistic view of accounting because the computer can process a much larger number of transactions than would be practical using a manual approach. Printouts produced by the programs can be used to effectively trace the flow of an accounting transaction through the accounting cycle. Financial statement analysis can be expanded since all information necessary for analysis is presented on the printout from the computer. Bank reconciliations, statement of change in financial condition, and other *detailed* analysis can easily be produced from the information printed by the computer. This additional analysis tends to give the student a better understanding of the accounting process.

### Advantages Derived from Integrating Computerized Concepts into Principles of Accounting

This computerized approach to the teaching of principles of accounting at Danville Community College presents the following advantages to the accounting student:

1. The student is relieved of tedious calculations—almost all arithmetic operations are performed by the computer.

2. The boredom of the posting process is eliminated.

3. The student has a complete set of books for analysis.

4. The student is introduced to accounting methodology which closely parallels actual business practice.

In addition, the use of the accounting package developed at DCC allows the instructor to quickly locate student errors in problems through the diagnostics produced by the programs. This additional time can be used to assist students in acquiring mastery of accounting principles rather than looking for arithmetic errors.

### Note

All correspondence and requests for program data should be directed to Professor Parrish.

**Figure 1**

The trial balance of Jones Office Supply Company at October 1 of the current year is presented below.

| Account No. | Account Name | Debit | Credit |
|---|---|---|---|
| 111 | Cash on hand | 100.00 | |
| 112 | Cash in bank | 7 564.44 | |
| 113 | Notes Receivable | 400.00 | |
| 114 | Accounts Receivable | 10 918.60 | |
| 117 | Merchandise Inventory | 7 500.00 | |
| 119 | Office supplies | 875.00 | |
| 120 | Prepaid Insurance | 600.00 | |
| 124 | Store Equipment | 25 000.00 | |
| 211 | Accounts Payable | | 340.00 |
| 218 | Sales Tax Payable | | 790.88 |
| 223 | Mortgage Payable | | 32 500.00 |
| 311 | A Capital | | 9 035.00 |
| 411 | Sales Department A | | 4 852.00 |
| 412 | Sales Department B | | 2 354.00 |
| 413 | Sales Department C | | 1 396.00 |
| 414 | Sales Department D | | 3 670.00 |
| 415 | Sales Discount | 87.08 | |
| 416 | Sales Returns and Allowance | 25.00 | |
| 418 | Interest Income | | 12.00 |
| 511 | Purchases | 1 300.00 | |
| 512 | Purchases Discounts | | 26.00 |
| 517 | Advertising Expense | 155.00 | |
| 520 | Store Supplies | 100.00 | |
| 542 | Repairs and Maintenance | 350.76 | |
| | Total | 54 975.88 | 54 975.88 |

The following transactions were completed by the Jones Office Supply Company during October of the current year.

Oct. 1 Sold merchandise on account to Bass Brothers, Inc., invoice No. 101, Department A, $580.00.

2 Sold merchandise on account to Cone Mfg. Co., invoice no. 102, Department B, $670.00 plus freight prepaid $15.00.

3 Purchased merchandise on account from Custom Wholesale Co., $126.10; order no. 201.

4 Purchased merchandise on account from Gregory Paper Co., $12,672.00, order no. 202.

5 Issued check no. 801 to Jones Realty in payment of rent for October, $400.00.

9 Received payment from Bass Brothers, Inc., for sales invoice no. 101, dated October 1 in the amount of $603.20 less 2 percent discount, receipt number 402, $580.00 sales price.

12 Issued check no. 802 to Custom Wholesale Co. in payment of purchase order no. 131 dated October 3, in the amount of $126.10, less 1 percent discount.

31 Receipt no. 403 indicated the following summary of cash sales for October 1-31.

| | Sales | Sales Tax Collected | Total |
|---|---|---|---|
| Dept. A | $1,573.00 | $62.92 | $1,635.92 |
| Dept. B | 460.00 | 18.40 | 478.40 |

31 Deposits to the bank for the month of October totaled $2,705.92. The total bank deposit must be recorded by a general journal entry with a debit to Account 112, cash in bank, and a credit to Account 111, cash on hand.

31 Recorded in the general journal the following payroll for October.

| | | |
|---|---|---|
| Sales Salaries | $2,050.00 | |
| Office Salaries | 450.00 | |
| Total | | $2,500.00 |
| Total Withheld: | | |
| Federal Income taxes | 275.00 | |
| FICA Taxes | 146.25 | |
| State Income Taxes | 75.00 | |
| Total | | 496.25 |
| Net Payroll | | $2,003.75 |

31 Issued check no. 803 to the Payroll Bank Account for the net amount of the payroll.

31 Issued purchase order no. 203 to Bell Telephone Co. for October telephone charges. $38.60.

**Figure 2**

## CHART OF ACCOUNTS

### Balance Sheet Accounts

1. Assets

| Account # | Account |
|---|---|
| 110 | Petty Cash |
| 111 | Cash on Hand |
| 112 | Cash in Bank |
| 113 | Notes Receivable |
| 114 | Accounts Receivable |
| 115 | Allowance for Doubtful Accts. |
| 116 | Marketable Securities |
| 117 | Merchandise Inventory |
| 118 | Store Supplies |
| 119 | Office Supplies |
| 120 | Prepaid Insurance |
| 121 | Prepaid Rent |
| 122 | Other Prepaid Items |
| 123 | Cash Value Life Insurance |
| 124 | Store Equipment |
| 125 | Accum. Depr. Store Equipment |
| 126 | Office Equipment |
| 127 | Accum Depr. Office Equipment |
| 128 | Auto and Trucks |
| 129 | Accum. Depr. Auto & Trucks |
| 130 | Buildings |
| 131 | Accum. Depr. - Buildings |
| 132 | Land |
| 133 | Goodwill |
| 134 | Organization Costs |
| 135 | Patents |
| 136 | Disc. on Bonds Payable |
| 137 | Other Assets |

2. Liabilities

| | |
|---|---|
| 211 | Accounts Payable |
| 212 | Notes Payable |
| 213 | Fed. Inc. Tax Withheld |
| 214 | FICA Tax Payable |
| 215 | State Inc. Tax Withheld |
| 216 | Fed. Unempl. Tax Payable |
| 217 | State Unempl. Tax Payable |
| 218 | Sales Tax Payable |
| 219 | Interest Payable |
| 220 | Dividends Payable |
| 221 | Salaries Payable |
| 222 | Accrued Liabilities |
| 223 | Mortgage Payable - Current |
| 224 | Bonds Payable |
| 230 | Deferred Taxes |
| 231 | Other Liabilities |

3. Capital

| | |
|---|---|
| 311 | A Capital |
| 312 | B Capital |
| 313 | A Drawing |
| 314 | B Drawing |
| 315 | Common Stock |
| 316 | Premium on Common Stock |
| 317 | Discount on Common Stock |
| 318 | Preferred Stock |
| 319 | Premium on Preferred Stock |
| 320 | Treasury Stock |
| 321 | Paid in Capital - Treas. Stock |
| 322 | Stock Dividend Distributable |
| 323 | Cash or Stock Dividend |
| 325 | Retained Earnings Appropriated |
| 326 | Retained Earnings Appropriated |
| 327 | Retained Earnings Appropriated |
| 328 | Retained Earnings - Unappropriated |
| 330 | Expense & Revenue Summary |

Figure 2 (con't.)

## Income Statement Accounts

4. Revenue

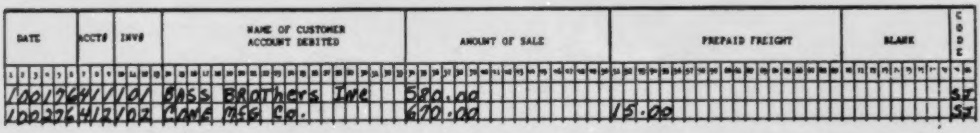
| Account # | Account |
|---|---|
| 411 | Sales Dept. A |
| 412 | Sales Dept. B |
| 413 | Sales Dept. C |
| 414 | Sales Dept. D |
| 415 | Sales Discounts |
| 416 | Sales Returns & Allowances |
| 417 | Service Revenue |
| 418 | Interest Income |
| 419 | Other Income |
| 420 | Gain on Sale of Assets |

5. Expenses

| | |
|---|---|
| | COST OF GOODS SOLD |
| 511 | Purchases |
| 512 | Purchase Discounts |
| 513 | Purchase Returns & Allowances |
| 514 | Freight In |
| | SELLING EXPENSES |
| 516 | Sales Salaries |
| 517 | Advertising Expenses |
| 518 | Depreciation - Store Equip. |
| 519 | Insurance - Selling |
| 520 | Store Supplies |
| 521 | Misc. Selling Expense |
| 522 | Delivery Expense |
| | GENERAL & ADMINISTRATIVE |
| 530 | Office Salaries |
| 531 | Officer Salaries |
| 532 | Taxes - Payroll |
| 533 | Taxes - Other |
| 534 | Depr. Exp. - Building |
| 535 | Depr. Exp. - Office |
| 536 | Ins. Exp. - General |
| 537 | Office Supplies Expense |
| 538 | Misc. General Expenses |
| 539 | Rent Expense |
| 540 | Utilities |
| 541 | Telephone |
| 542 | Repairs & Maintenance |
| 543 | Bad Debt Expense |
| 550 | Interest Expense |
| 551 | Loss on Sale of Assets |

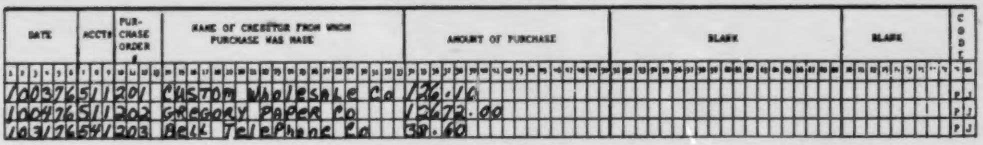
Figure 3

SALES JOURNAL CODING FORM



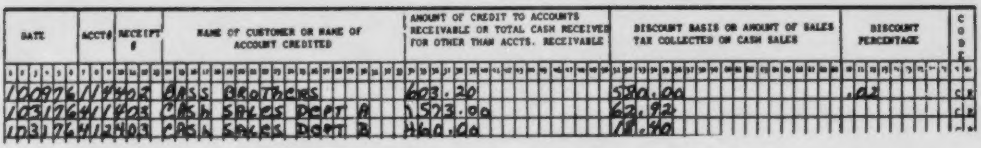
| DATE | ACCT# | INV# | NAME OF CUSTOMER ACCOUNT DEBITED | AMOUNT OF SALE | PREPAID FREIGHT | BLANK | CODE |
|---|---|---|---|---|---|---|---|
| 100176 | 411 | 101 | BASS BROThers Inc | 580.00 | | | SJ |
| 100276 | 412 | 102 | CAME MFG Co. | 670.00 | 15.00 | | SJ |

PURCHASES JOURNAL OR VOUCHER REGISTER



| DATE | ACCT# | PURCHASE ORDER # | NAME OF CREDITOR FROM WHOM PURCHASE WAS MADE | AMOUNT OF PURCHASE | BLANK | BLANK | CODE |
|---|---|---|---|---|---|---|---|
| 100376 | 511 | 201 | CUSTOM WHolESALE Co | 126.10 | | | PJ |
| 100476 | 511 | 202 | GREGORY PAPER Co | 12672.00 | | | PJ |
| 103176 | 541 | 203 | BELL TElePhone Co | 38.60 | | | PJ |

CASH RECEIPTS JOURNAL



| DATE | ACCT# | RECEIPT # | NAME OF CUSTOMER OR NAME OF ACCOUNT CREDITED | AMOUNT OF CREDIT TO ACCOUNTS RECEIVABLE OR TOTAL CASH RECEIVED FOR OTHER THAN ACCTS. RECEIVABLE | DISCOUNT BASIS OR AMOUNT OF SALES TAX COLLECTED ON CASH SALES | DISCOUNT PERCENTAGE | CODE |
|---|---|---|---|---|---|---|---|
| 100976 | 114 | 402 | BASS BROThers | 603.20 | 580.00 | .02 | CR |
| 103176 | 411 | 403 | CASH SALES DEPT A | 1573.00 | 62.92 | | CR |
| 103176 | 412 | 403 | CASH SALES DEPT B | 460.00 | 18.40 | | CR |

Figure 3 (con't.)

CASH PAYMENTS JOURNAL OR CHECK REGISTER

| DATE | ACCT# | CHECK# | PAYEE OF CHECK | AMOUNT OF DEBIT TO ACCOUNTS PAYABLE OR TOTAL CASH PAID FOR OTHER THAN ACCOUNTS PAYABLE | DISCOUNT BASIS | DISCOUNT PERCENTAGE | CODE |
|---|---|---|---|---|---|---|---|
| 100576 | 539 | 801 | Jones Realty | 400.00 | | | CP |
| 101276 | 211 | 802 | Custom Wholesale Co | 126.10 | 126.10 | .01 | CP |
| 103176 | 221 | 803 | Payroll Bank Account | 2003.75 | | | CP |
| | | | | | | | CP |
| | | | | | | | CP |
| | | | | | | | CP |
| | | | | | | | CP |

GENERAL JOURNAL

| DATE | ACCT # | | THE NAME OF THE ACCOUNT WHICH IS DEBITED OR CREDITED. | AMOUNT OF DEBIT OR CREDIT | BLANK | Code |
|---|---|---|---|---|---|---|
| 103176 | 112 | | Cash in Bank | 2705.92 | | GJ |
| 103176 | 111 | | Cash on Hand | -2705.92 | | GJ |
| 103176 | 516 | | Sales Salaries | 2050.00 | | GJ |
| 103176 | 530 | | Office Salaries | 450.00 | | GJ |
| 103176 | 213 | | Federal Income Tax | -275.00 | | GJ |
| 103176 | 214 | | FICA Tax Withheld | -146.25 | | GJ |
| 103176 | 215 | | State Income Tax | -75.00 | | GJ |
| 103176 | 221 | | Salaries Payable | -2003.75 | | GJ |

**Figure 4**

```
                                   ***SALES JOURNAL***                                            PAGE  1

                                             ACCOUNTS    SALES    DELIVERY    SALES     SALES     SALES     SALES
                                  INVOICE      REC        TAX     EXPENSE    DEPT. A   DEPT. B   DEPT. C   DEPT. D
DATE    ACCOUNT DEBITED           NUMBER      (DR)       (CR)      (CR)       (CR)      (CR)      (CR)      (CR)
---------------------------------------------------------------------------------------------------------------------

10- 1   BASS BROTHERS, INC          101      603.20     23.20                580.00

10- 2   CONE MFG CO                 102      711.80     26.80     15.00                670.00
                                            --------  --------  --------  --------  --------  --------  --------

                                            1315.00     50.00     15.00    580.00    670.00       0.0       0.0
                                            ========  ========  ========  ========  ========  ========  ========
```

**Figure 5**

```
                                 ***PURCHASES JOURNAL***                                           PAGE  1

                                      ACCOUNTS                STORE      OFFICE     SUNDRY ACCOUNTS (DR)
                             ORDER    PAYABLE    PURCHASES   SUPPLIES   SUPPLIES
DATE   ACCOUNT CREDITED      NUMBER    (CR)        (DR)        (DR)       (DR)    ACCOUNT                    AMOUNT
---------------------------------------------------------------------------------------------------------------------

10- 3  CUSTOM WHOLESALE CO    201       126.10      126.10

10- 4  GREGORY PAPER CO       202     12672.00    12672.00

10-31  BELL TELEPHONE CO      203        38.60                                      TELEPHONE EXPENSE            38.60
                                      --------    --------    --------   --------                           --------

                                      12836.70    12798.10       0.0        0.0                                  38.60
                                      ========    ========    ========   ========                           ========
```

**Figure 6**

```
                                ***CASH RECEIPTS JOURNAL***                                        PAGE  1

                                  SUNDRY    SALES    SALES    SALES    SALES    SALES   ACCOUNTS   SALES
                        RECEIPT  ACCOUNTS  DEPT. A  DEPT. B  DEPT. C  DEPT.D     TAX      REC    DISCOUNT    CASH
DATE   ACCOUNT CREDITED NUMBER    (CR)      (CR)     (CR)     (CR)     (CR)     (CR)      (CR)     (CR)      (DR)
---------------------------------------------------------------------------------------------------------------------

10- 9  BASS BROTHERS     402                                                            603.20    11.60    591.60

10-31  CASH SALES DEPT A 403              1573.00                               62.92                      1635.92

10-31  CASH SALES DEPT B 403                        460.00                      18.40                       478.40
                                 -------- -------- -------- -------- -------- -------- -------- -------- --------

                                     0.0  1573.00   460.00      0.0      0.0    81.32   603.20    11.60   2705.92
                                 ======== ======== ======== ======== ======== ======== ======== ======== ========
```

Figure 7

```
                                  ***CASH PAYMENTS JOURNAL***

                                                SUNDRY     ACCOUNTS     PURCHASES
                                     CHECK     ACCOUNTS     PAYABLE      DISCOUNT        CASH
DATE    ACCOUNT DEBITED              NUMBER      (DR)         (DR)         (CR)          (CR)
-----------------------------------------------------------------------------------------------------

10- 5   JONES REALTY                  801       400.00                                 400.00

10-12   CUSTOM WHOLESALE CO           802                   126.10        1.26         124.84

10-31   PAYROLL BANK ACCOUNT          803      2003.75                                2003.75
                                               --------     --------     --------     --------

                                               2403.75       126.10        1.26       2528.59
                                               ========     ========     ========     ========
```

Figure 8

```
                         ***GENERAL JOURNAL***

DATE       ACCOUNT                                DEBIT        CREDIT
-------------------------------------------------------------------------

10-31      CASH IN BANK                          2705.92
                CASH ON HAND                                   2705.92

10-31      SALES SALARIES                        2050.00
           OFFICE SALARIES                        450.00
                FED INC TAX WITHHELD                            275.00
                FICA TAX PAYABLE                                146.25
                STATE INC TAX WITH                               75.00
                SALARIES PAYABLE                               2003.75
                                              ----------    ----------
           TOTAL                                 5205.92       5205.92
                                              ==========    ==========
```

Figure 9

```
                          ***GENERAL LEDGER***

ACCOUNT NAME:  CASH ON HAND                             ACCOUNT NO.  111
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE               100.00                  100.00
10- 1     CR                             2705.92                 2305.92
10-31     GJ                                          2705.92     100.00

ACCOUNT NAME:  CASH IN BANK                             ACCOUNT NO.  112
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE              7564.44                 7564.44
10- 1     CP                                          2528.59    5035.85
10-31     GJ                             2705.92                 7741.77

ACCOUNT NAME:  NOTES RECEIVABLE                         ACCOUNT NO.  113
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE               400.00                  400.00

ACCOUNT NAME:  ACCOUNTS RECEIVAB                        ACCOUNT NO.  114
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE             10918.60                10918.60
10- 1     SJ                             1315.00                12233.60
10- 1     CR                                           603.20   11630.40

ACCOUNT NAME:  MERCHANDISE INVENT                       ACCOUNT NO.  117
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE              7500.00                 7500.00

ACCOUNT NAME:  OFFICE SUPPLIES                          ACCOUNT NO.  119
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE               875.00                  875.00

ACCOUNT NAME:  PREPAID INSURANCE                        ACCOUNT NO.  120
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE               600.00                  600.00

ACCOUNT NAME:  STORE EQUIPMENT                          ACCOUNT NO.  124
------------------------------------------------------------------------
DATE      REFERENCE                        DEBIT     CREDIT      BALANCE
------------------------------------------------------------------------
10- 1     BEGINNING BALANCE             25000.00                25000.00
```

Figure 10

```
                          ***TRIAL BALANCE***

NO   ACCOUNT                              DEBIT              CREDIT
-----------------------------------------------------------------------
111  CASH ON HAND                        100.00
112  CASH IN BANK                       7741.77
113  NOTES RECEIVABLE                    400.00
114  ACCOUNTS RECEIVAB                 11630.40
117  MERCHANDISE INVENT                 7500.00
119  OFFICE SUPPLIES                     875.00
120  PREPAID INSURANCE                   600.00
124  STORE EQUIPMENT                   25000.00
211  ACCOUNTS PAYABLE                                      13050.60
213  FED INC TAX WITHHELD                                    275.00
214  FICA TAX PAYABLE                                        146.25
215  STATE INC TAX WITH                                       75.00
218  SALES TAX PAYBL                                         922.20
223  MORTGAGE PAYABLE                                      32500.00
311  A CAPITAL                                              9035.00
411  SALES DEPT A                                           7005.00
412  SALES DEPT B                                           3484.00
413  SALES DEPT C                                           1396.00
414  SALES DEPT D                                           3670.00
415  SALES DISCOUNT                       98.68
416  SALES RET AND ALLOWS                 25.00
419  INTEREST INCOME                                          12.00
511  PURCHS                            14098.10
512  PURCHS DISCTS                                            27.26
516  SALES SALARIES                     2050.00
517  ADVERTISING EXPENSE                 155.00
520  STORE SUPPLIES                      100.00
522  DELIVERY EXPENSES                                        15.00
530  OFFICE SALARIES                     450.00
539  RENT EXPENSE                        400.00
541  TELEPHONE EXPENSE                    38.60
542  REPAIRS AND MAINTEN                 350.76
                                     ----------          ----------

                                       71613.31            71613.31
                                     ==========          ==========
```

Figure 11

An examination of the trial balance prepared in Phase I reveals the necessity for the following adjustments before financial statements are prepared:

1. A physical count of the merchandise inventory at October 31 revealed the value of merchandise inventory to be $14,589.50.
2. The value of office supply inventory is $526.50.
3. Insurance expired during the month was $50.00.
4. Depreciation on store equipment for the month was $49.78.
5. Employer payroll taxes for the month are as follows:

| | |
|---|---|
| FICA Expense | $146.25 |
| Federal Unemployment | 67.50 |
| State Unemployment | 10.00 |
| Total | $223.75 |

6. Interest accrued on Mortgage Payable. $208.93
7. Allowance for doubtful accounts is estimated at 1% of the balance in Accounts Receivable.

Figure 11-A

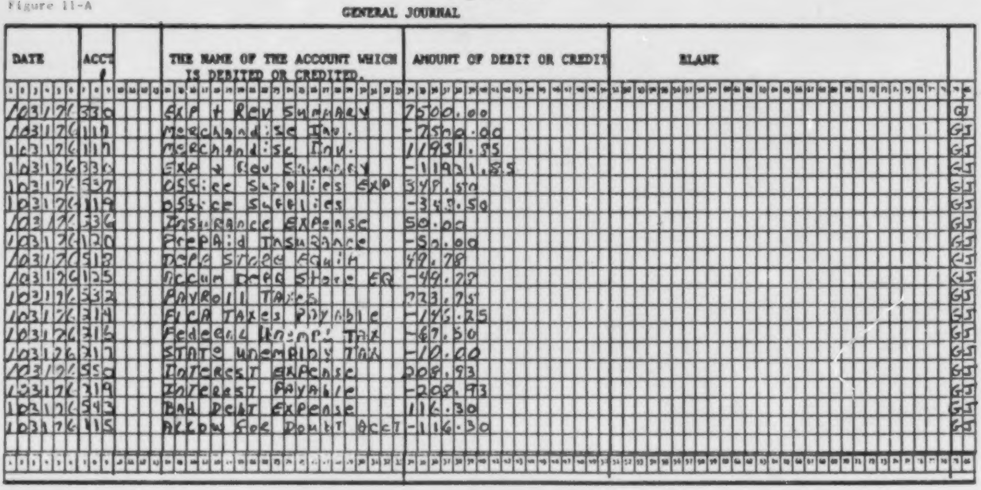


Figure 11-A

GENERAL JOURNAL

| DATE | ACCT # | | THE NAME OF THE ACCOUNT WHICH IS DEBITED OR CREDITED. | AMOUNT OF DEBIT OR CREDIT | BLANK | |
|---|---|---|---|---|---|---|
| 103176 | 330 | | Exp + Rev Summary | 7500.00 | | GJ |
| 103176 | 110 | | Merchandise Inv. | -7500.00 | | GJ |
| 103176 | 110 | | Merchandise Inv. | 11931.85 | | GJ |
| 103176 | 330 | | Exp + Rev Summary | -11931.85 | | GJ |
| 103176 | 537 | | Office Supplies Exp | 348.50 | | GJ |
| 103176 | 119 | | Office Supplies | -348.50 | | GJ |
| 103176 | 536 | | Insurance Expense | 50.00 | | GJ |
| 103176 | 120 | | Prepaid Insurance | -50.00 | | GJ |
| 103176 | 518 | | Depr Store Equip | 49.78 | | GJ |
| 103176 | 125 | | Accum Depr Store EQ | -49.78 | | GJ |
| 103176 | 532 | | Payroll Taxes | 223.75 | | GJ |
| 103176 | 314 | | FICA Taxes Payable | -146.25 | | GJ |
| 103176 | 316 | | Federal Unempl Tax | -67.50 | | GJ |
| 103176 | 317 | | State Unemploy Tax | -10.00 | | GJ |
| 103176 | 550 | | Interest Expense | 208.93 | | GJ |
| 103176 | 319 | | Interest Payable | -208.93 | | GJ |
| 103176 | 543 | | Bad Debt Expense | 116.30 | | GJ |
| 103176 | 115 | | Allow For Doubt Acct | -116.30 | | GJ |

Figure 12

```
                            ***GENERAL JOURNAL***
DATE       ACCOUNT                                  DEBIT          CREDIT
-------------------------------------------------------------------------
                             ADJUSTING ENTRIES
10-31      EXP & REV SUMMARY                       7500.00
                MERCHANDISE INVENT                                7500.00

10-31      MERCHANDISE INVENT                     11931.85
                EXP & REV SUMMARY                                11931.85

10-31      OFFICE SUPPLIES EXP                      348.50
                OFFICE SUPPLIES                                    348.50

10-31      INSURANCE EX GENER                        50.00
                PREPAID INSURANCE                                   50.00

10-31      DEPR STOR EQUIP                           49.78
                ACCUM DEPR STO EQUIP                                49.78

10-31      PAYROLL TAXES                            223.75
                FICA TAX PAYABLE                                   146.25
                FED UNEM TAX PAY                                    67.50
                STATE UNEM TAX PAY                                  10.00

10-31      INTEREST EXPENSE                         208.93
                INTEREST PAYABLE                                   208.93

10-31      BAD DEBT EXPENSE                         116.30
                ALLOW FOR DOUBT ACCT                               116.30
                                                ----------     ----------
           TOTAL                                  20429.11       20429.11
                                                ==========     ==========
```

Figure 12-A

```
ACCOUNT NAME:  ALLOW FOR DOUBT ACCT                       ACCOUNT NO.  115
--------------------------------------------------------------------------
DATE       REFERENCE                        DEBIT        CREDIT      BALANCE
--------------------------------------------------------------------------

10-30      BEGINNING BALANCE                                             0.0
10-31      GJ                                           116.30       -116.30



ACCOUNT NAME:  MERCHANDISE INVENT                         ACCOUNT NO.  117
--------------------------------------------------------------------------
DATE       REFERENCE                        DEBIT        CREDIT      BALANCE
--------------------------------------------------------------------------

10-30      BEGINNING BALANCE              7500.00                    7500.00
10-31      GJ                                          7500.00          0.0
10-31      GJ                            11931.85                   11931.85



ACCOUNT NAME:  OFFICE SUPPLIES                            ACCOUNT NO.  119
--------------------------------------------------------------------------
DATE       REFERENCE                        DEBIT        CREDIT      BALANCE
--------------------------------------------------------------------------

10-30      BEGINNING BALANCE               875.00                     875.00
10-31      GJ                                           348.50        526.50



ACCOUNT NAME:  PREPAID INSURANCE                          ACCOUNT NO.  120
--------------------------------------------------------------------------
DATE       REFERENCE                        DEBIT        CREDIT      BALANCE
--------------------------------------------------------------------------

10-30      BEGINNING BALANCE               600.00                     600.00
10-31      GJ                                            50.00        550.00



ACCOUNT NAME:  STORE EQUIPMENT                            ACCOUNT NO.  124
--------------------------------------------------------------------------
DATE       REFERENCE                        DEBIT        CREDIT      BALANCE
--------------------------------------------------------------------------

10-30      BEGINNING BALANCE             25000.00                   25000.00
```

**Figure 13**

```
                         ***ADJUSTED TRIAL BALANCE***

NO    ACCOUNT                                     DEBIT                 CREDIT
-------------------------------------------------------------------------------
111    CASH ON HAND                              100.00
112    CASH IN BANK                             7741.77
113    NOTES RECEIVABLE                          400.00
114    ACCOUNTS RECEIVAB                       11630.40
115    ALLOW FOR DOUBT ACCT                                             116.30
117    MERCHANDISE INVENT                      11931.85
119    OFFICE SUPPLIES                           526.50
120    PREPAID INSURANCE                         550.00
124    STORE EQUIPMENT                         25000.00
125    ACCUM DEPR STO EQUIP                                              49.78
211    ACCOUNTS PAYABLE                                               13050.60
213    FED INC TAX WITHHELD                                             275.00
214    FICA TAX PAYABLE                                                 292.50
215    STATE INC TAX WITH                                                75.00
216    FED UNEM TAX PAY                                                  67.50
217    STATE UNEM TAX PAY                                                10.00
218    SALES TAX PAYBL                                                  922.20
219    INTEREST PAYABLE                                                 208.93
223    MORTGAGE PAYABLE                                               32500.00
311    A CAPITAL                                                       9035.00
330    EXP & REV SUMMARY                                               4431.85
411    SALES DEPT A                                                    7005.00
412    SALES DEPT B                                                    3484.00
413    SALES DEPT C                                                    1396.00
414    SALES DEPT D                                                    3670.00
415    SALES DISCOUNT                             98.68
416    SALES RET AND ALLOWS                       25.00
418    INTEREST INCOME                                                   12.00
511    PURCHS                                  14098.10
512    PURCHS DISCTS                                                     27.26
516    SALES SALARIES                           2050.00
517    ADVERTISING EXPENSE                       155.00
518    DEPR STOR EQUIP                            49.78
520    STORE SUPPLIES                            100.00
522    DELIVERY EXPENSES                                                 15.00
530    OFFICE SALARIES                           450.00
532    PAYROLL TAXES                             223.75
536    INSURANCE EX GENER                         50.00
537    OFFICE SUPPLIES EXP                       348.50
539    RENT EXPENSE                              400.00
541    TELEPHONE EXPENSE                          38.60
542    REPAIRS AND MAINTEN                       350.76
543    BAD DEBT EXPENSE                          116.30
550    INTEREST EXPENSE                          208.93
                                              ----------             ----------

                                                76643.92               76643.92
                                              ==========             ==========
```

**Figure 14**

```
                              ***INCOME STATEMENT***

 REVENUE:
      SALES DEPT A                             7005.00
      SALES DEPT B                             3484.00
      SALES DEPT C                             1396.00
      SALES DEPT D                             3670.00
                                             ---------
        TOTAL SALES
      LESS: SALES RET AND ALLOWS                           15555.00
      LESS: SALES DISCOUNT                       25.00
                                                 98.68
                                             ---------
                                                             123.68
      NET SALES/REVENUE                                   ---------
                                                                      15431.32
 COST OF MERCHANDISE SOLD:
      BEGINNING INVENTORY                                   7500.00
      PURCHS                                  14098.10
                                             ---------
        TOTAL                                 14098.10
      LESS: PURCHS DISCTS                        27.26
                                             ---------
      NET PURCHASES
                                                           14070.84
      MDSE AVAILABLE FOR SALE                             ---------
      LESS:  ENDING INVENTORY                              21570.84
                                                           11931.85
        COST OF MDSE SOLD                                 ---------
                                                                       9638.99
                                                                     ---------
 GROSS PROFIT ON SALES
                                                                       5792.33
 OPERATING EXPENSES:
      SELLING EXPENSES:
        SALES SALARIES
        ADVERTISING EXPENSE                    2050.00
        DEPR STOR EQUIP                         155.00
        STORE SUPPLIES                           49.78
        DELIVERY EXPENSES                       100.00
                                                -15.00
                                             ---------
          TOTAL SELLING EXPENSES
                                                            2339.78
      GENERAL EXPENSES:
        OFFICE SALARIES
        PAYROLL TAXES                           450.00
        INSURANCE EX GENER                      223.75
        OFFICE SUPPLIES EXP                      50.00
        RENT EXPENSE                            348.50
        TELEPHONE EXPENSE                       400.00
        REPAIRS AND MAINTEN                      38.60
        BAD DEBT EXPENSE                        350.76
                                                116.30
                                             ---------
          TOTAL GENERAL EXPENSES
                                                            1977.91
                                                          ---------
     TOTAL OPERATING EXPENSES
                                                                       4317.69
                                                                     ---------
 INCOME FROM OPERATIONS
                                                                       1474.64
 OTHER INCOME:
      INTEREST INCOME
                                                 12.00
                                             ---------
         TOTAL OTHER INCOME
                                                              12.00
 OTHER EXPENSES:
      INTEREST EXPENSE                          208.93
                                             ---------
         TOTAL OTHER EXPENSES
                                                             208.93
                                                          ---------
                                                                       -196.93
 NET INCOME BEFORE TAXES                                             ---------
                                                                       1277.71
                                                                     =========
```

Figure 15

```
                       ***CAPITAL STATEMENT***

BEGINNING A CAPITAL                                      9035.00
NET INCOME                                   1277.71
    LESS: A DRAWING                              0.0
                                           ---------
NET INCREASE                                             1277.71
                                                       ---------
ENDING CAPITAL                                          10312.71
                                                       =========
```

Figure 16

```
                               ***BALANCE SHEET***

                                     ASSETS
                                     ------

CURRENT ASSETS:
  CASH ON HAND                                              100.00
  CASH IN BANK                                             7741.77
  NOTES RECEIVABLE                            400.00
  ACCOUNTS RECEIVAB                         11630.40
    LESS:  ALLOW FOR DOUBT ACCT              -116.30
                                           ---------
                                                          11914.10
  MERCHANDISE INVENT                                      11931.85
  OFFICE SUPPLIES                                           526.50
  PREPAID INSURANCE                                         550.00
                                                         ---------
                                                                      32764.22

LAND, BUILDINGS, AND EQUIPMENT:
                                          ACCUMULATED      BOOK
                                COST      DEPRECIATION     VALUE
  STORE EQUIPMENT             25000.00         49.78     24950.22
                             ---------     ---------     ---------
                              25000.00         49.78                  24950.22
                             =========     =========
                                                                     ---------

TOTAL ASSETS                                                          57714.44
                                                                     =========

                                  LIABILITIES
                                  -----------

CURRENT LIABILITIES:
  ACCOUNTS PAYABLE                                        13050.60
  FED INC TAX WITHHELD                                      275.00
  FICA TAX PAYABLE                                          292.50
  STATE INC TAX WITH                                         75.00
  FED UNEM TAX PAY                                           67.50
  STATE UNEM TAX PAY                                         10.00
  SALES TAX PAYBL                                           922.20
  INTEREST PAYABLE                                          208.93
                                                         ---------
                                                                      14901.73

LONG-TERM DEBT:
  MORTGAGE PAYABLE                                        32500.00
                                                         ---------
                                                                      32500.00
                                                                     ---------

TOTAL LIABILITIES                                                     47401.73

                                   CAPITAL
                                   -------

ENDING CAPITAL                                                        10312.71

TOTAL LIABILITIES AND CAPITAL                                         57714.44
                                                                     =========
```

# Computer-Assisted Learning
# Application to Planning in Command Economies

**William L. Henderson**
*Department of Economics*
*Denison University*
*Granville, Ohio 43023*
*(614) 587-0810 Ext. 473*

## Introduction

The development of this computer-assisted learning activity grew out of a practical need to improve source materials for a course in Comparative Economic Systems. The support for the activity was provided by the Denison Simulation Center. Denison University has been aided in the development of computer related learning, primarily simulation modeling, by a grant from Lilly Endowment. Most of the funds have been made available to faculty for more traditional simulation projects. The author's activities were devoted to a somewhat different alternative to the lecture delivery system. The project was designed to provide a computer-generated description of planning and simulation of the planners' administration of command economies.

## The Nature of the Learning Problem

The above comments are instructive with reference to the author's view that scope and format of available source materials represented a problem. A problem of coverage and appropriate emphasis is inherent in any survey course which includes a large number of topics and/or areas. In this course, students experienced some problems in understanding concepts without additional and disjunctive coverage of specialized topics. Also, the traditional lecture format was less than effective for particular sections.

## The Source Materials for the Course

The literature and source materials in the field are available in two forms: (1) survey texts, and (2) journal articles on specialized topics. The survey texts are encyclopedic in nature. These texts typically begin with an introduction to Marxian and capitalistic ideology. Materials are then developed to provide historical surveys of political-economic organization and operation of East Europe, the USSR, China, and other economies. Most texts include a chapter or section on economic planning, its structure, administration, and operation.

The article source materials are found in specialized journals. The cumulative requirements for coverage of appropriate topics represents a challenge in terms of the students' reading time.

Either form of the source materials means fast paced coverage of essential subject matter. Vital subject matter necessitates coverage of ideology, economic organization and operation of West European economies as well as the other Eastern systems noted above. So, two texts were assigned for all students. Reading assignments included the sections on planning.

## Focus on Planning

A ubiquitous feature of the economies (W. Germany excepted) is government administration of the economy. The principal form of control is, of course, public ownership of resources and production facilities. An operational element in the administration of these economies is the use of economic planning. The extent, type, and administration of the planning processes vary between and among the nations. But a basic understanding of the planning processes requires information about the economies as well as an understanding of analytical tools. It is the centricity of planning and the nature of the related tools that provided the impetus to work on alternative delivery systems.

There was a problem of providing adequate information on the mechanical processes without consuming a disproportionate amount of time in explaining the tools of analysis. To help the students understand the means of allocation control in planned systems, it was important to develop explanations of methods by which the information is managed by staff statistical agencies of the planning boards. There are a number of good texts available on analytical tools such as input-output, linear programming, and related tools. But due to the dimensions of the course and variegated student preparation, working through these texts was counterproductive. Thus, there was a need to explain the tools without distorting other more important course objectives.

The modification in the traditional reading, lecture format was a computer delivery or exposition of planning processes in planned economies. The following paragraphs summarize the development, use, and partial evaluation of the computer produced cognitive learning exercise.

### Computer-Produced Subject Matter

The computer program provides: (1) basic data on the political structure of economic administration in a command economy, (2) basic information on the construction and application of input-output analysis, (3) the use of the input-output organizing system for determining the impact of decisions throughout all sectors of a planned economy, and (4) simulated activities of the planners in commanding production and allocation.

The computer mechanical processes are used in a symbolic way to illustrate the dead weight burden of planning administration and the communication requirements of an extensive, sophisticated loop feedback system.

### Input-Output Analysis (in Brief)

It may be helpful to understand how the basic subject matter of input-output analysis may interdict the lecture delivery method.

Input-output analysis was initiated some thirty years ago by a Nobel laureate economist as a tool to replicate the interdependence in resource allocation. The operational form of input-output is a series of tables showing rows and columns of numbers. The matrices provide information on the relationships of users and producers of goods. The matrices identify how the output of each or selected industries or sectors is allocated or distributed within the sector or industry and to other sectors or industries. Thus for any sector, the use of the output can be identified. The matrix can provide the information to determine the interrelationships between resources available or used in existing, proposed or desirable production.

Transaction tables use rows and columns arranged to show the interrelationships between user and final demand. The rows show the ouput of each sector available to other industries. Each column shows the inputs used by each sector.

There is a family of tables used in input-output analysis. One type of table shows intermediate and final demand by sources and users of inputs. These data show the first round effects of changes in gross output of any one sector on the industries from which it receives outputs. Another type of table shows the indirect or cumulative effects on final demand of sectors or the total change in output in all industries resulting from additional units of output as final demand. My purpose is not to review the mechanics of input-output, but to explain the significance of "what is" presented in input-output tables.

If central planners dictate an increase in auto output of 10,000 vehicles, it is easy to identify first level changes in output required by the producing sectors. Ten thousand more cars means 50,000 new tires and wheels, 10,000 additional engine blocks, 40,000 to 80,000 new spark plugs, 10,000 new steering wheels, etc. Those additional production requirements are readily translated to the producing sectors by the planners. The planners are able to determine additional resource needs for the added car production.

Each of the producing sectors, however, has many sub-suppliers which have increased production goals. More coal, limestone, and iron ore are required by the steel industry to produce the additional engine blocks. The coal industry needs more factors to permit it to increase output to meet the new production goals. The impact of the new higher levels of car production will permeate the economy. Given existing short run capacity, it may be necessary for the steel sector to expand its own production capacity to produce needed new steel for more engine blocks. A new production line will require more steel to build factories to house the new steel production facilities, as well as to produce the new machine tools to produce the new blocks. This condition also may occur in the rubber industry, or any sector responsible for more output.

Thus the increase in auto production may require that many sectors increase their output by a multiple larger than apparent in the original new production requirements. The indirect coefficients in an input-output table produced by matrix inversion provide the planners with the necessary numbers to understand the cumulative impact of all user changes and all final output adjustments.

## The Importance of Input-Output Tools

There are two significant elements about the I-O analysis and the course delivery system. On the one hand, the concept is fundamentally important. It is a process for effective decision making by planners in many countries. It is a tool which has practical, everyday application in existing planned systems. This role of the analytical tool is essential to the students' understanding of economic administration where resources are owned and allocated by government agencies.

The significant second point is that the analysis or models are cumbersome to teach in the classroom-lecture format. If the models were developed as in mathematics classes, the time commitment would be extensive. The mathematical knowledge requirement also intercepts the logical prerequisities for the other course materials. Pragmatically, it would be impossible to illustrate the inversion of the large matrices or illustrate the required indirect coefficients.

## The Computer Role and Its Use

It was quite clear that in this case, computer-assisted instruction/simulation uniquely met specific classroom needs. Or stated differently, the particular course and subject matter have features that facilitated the application of computer instruction.

The project orientation was to provide a more practical way for students to understand the planning processes as actually carried on in planned economies. The delivery system was not a supplement to class lecture, but a substitute for explaining complex concepts without distorting the course with time consuming ancillary mechanical tools.

Before referring to the evaluation, it may be useful to comment on the use of the new delivery system. The planning coverage in the course appears midway through the term. The course organization permits about one week for discussion of planning. During the week, students are given discretion in opting for the computer-assisted learning and in scheduling the computer instruction.

There is no requirement for prior computer experience. The operating instructions may be followed in a step-by-step process to permit every student command over the machine operation. The user can determine the daily work commitment. Users can terminate at several places in the cumulative presentation of information.

Total running time depends, in part, upon the time used to respond to programmed questions.

The running time of the first section on political-economic organization is about fifteen minutes. That section provides an overview of the Soviet planning structure and administration.

Although the economy is called CHRUSNIYA, the organization, planning and operations replicate the Russian economy.

Student participation during this and other sections is limited to controlling the mechanics of printer operation and viewing the substantive materials that gurgle out of the printer, and answering questions at the end of the section. Each user must "punch in" replies to several inquiries.

The second section run takes about fifteen minutes to illustrate the operation or application of the input-output analysis. Direct and indirect coefficients were obtained from the 1966 USSR, sixty-two sector tables. Because of machine and printer capacity, it was necessary to compact the real world tables to twenty-six sectors. But the 26 by 26 sector tables retained actual coefficients from the Soviet input-output table. The numbers tnus illustrate a real world use of the analytical model. That section moves the student through the construction of a simple four sector illustrative table (see Appendix I) to the development of the twenty-six sector tables. The I-O tables are constructed to show actual and planned future coefficients. The data were presented to illustrate a 1975 economy with goals for a five-year plan that ends in 1980. The 1975 output and 1980 goals are stated in aggregated terms of gross value of output in a currency called "rubrincs." The student is informed that the projected sector output changes and the sectors selected for priority or rapid expansion of output.

At the end of this section, the user answers a series of questions about the matrix construction and the quantitative features of the planned increases in output.

The final section emphasizes the administration of the plan through the five-year period. There is emphasis on changes in plans and on the management of the information system. This section takes about forty minutes, plus time devoted to answering questions. Examples or illustrations include: (1) a downturn in agriculture production because of bad weather, (2) import substitution of automobiles, and (3) the effect of innovation in production techniques. In each case, the program emphasizes: (1) the use of the I-O analysis by tracing the chain of events through affected sectors, (2) the central control of resource allocation, and (3) the dead weight burden of communicating plan changes throughout the producing sectors.

For example, in import substitution, the domestic economy cannot achieve desired goals in auto production. The planners import cars, but the imports have higher gasoline consumption because of western pollution controls. The importation is a resource-saving decision for the sectors responsible for increased auto production. It is resource exhausting in the petroleum and related sectors. This example change in output and use is illustrated by the printing of changed coefficients for each affected sector (see Appendix II). The sector output effects are determined by the indirect coefficients.

## The Symbolic Use of the Computer

These illustrations are also used to symbolize the time constraints inherent in the information exchange required in planned systems. Each sector change requires the printing of a new 26 by 26 sector table. That is sheer "grunt" work. The machine illustrates this by a fairly long gurgling period to print the new numbers for the large tables. Each time an exigency occurs or other change in output is ordered, the planners must communicate with the managers of the affected sectors.

To symbolize the extensive communication process, a series of telegrams is printed (see Appendix III). The printing time is extended by the use of elaborate borders around each telegram. Symbolically, the volume of telegrams and the length of time required to print the messages illustrates the communication burden. In this learning process the computer and printer are used to replicate integral features of economic decision-making. The machine action illustrates the reliance upon machinery rather a large number of independent private decision makers. The machine time in tracing impacts through various sectors, illustrates the time consumption feature of top heavy administration. The time committed to reproducing large numbers of messages, illustrates the paper processing dimension of the communication loop feedback system that is an integral feature of planned systems. The user finds himself subservient to the machine. There are no perturbations introduced into the model by the user. That is also symbolic of decision control in planned economies. The student also becomes aware of the clerical syndrome as an observer by waiting and responding to slowly printed orders from some unidentified government decision-maker. Thus the planners dominance of the economy is simulated.

### Real World Relevance

The model was developed to reflect those real world aspects of the administration of an economic system. The model uses actual data and the input-output format from a real world economy. The simulation of the processes of production are based on real world events. The simulation of government or administrative decisions is an accurate reflection of processes that exist in planned economies. The action, decisions, and role of the planners in controlling allocation are simulated by the model.

### Student Evaluation and Impact Evidence

During the first semester 1976-77, the first student users were upper division social science majors. Testing the results of the learning format was initiated with that group. Control questions provide the only current basis for assessing cognitive effects. Two final test questions were designed to elicit responses about the nature of the planning process, and the interconnection in resource allocation in planned systems. The questions were asked of the twenty-five students who substituted the computer learning for the lecture and of the fifteen other students who did not opt to use the computer learning format.

There was little difference in the question scores on the planning process question between students who read the texts and the computer users. The computer users however recorded higher scores on the question on interrelatedness of resource allocation. The average test grade for the users was more than four percentage points above that for the non-users.

Student comments on the course questionnaire ranged from open enthusiasm for the computer model to a suggestion that all classroom presentation of the planning section be eliminated. At the time this is being written evaluation is ongoing. Regression will be used to identify significance of SAT, prior courses, and math preparation. However there is insufficient evidence to justify abandoning the format.

During the second semester 1976-77, the format will be used by freshmen in a module of the economics principles course and in an interdepartmental course on the contemporary Soviet Union, sponsored by a Russian language instructor. Each of the several new users will provide personal and academic data and complete a questionnaire designed to assess affective learning impact. There will be control questions on hourly exams to assess the cognitive features of the computer-learning format. The control questions will be related to re-test information. The significance of grade point hour ratios and prior course preparatin will be evaluated.

### Conclusion

The availability of the computer format alleviated selected problems in classroom presentation. This writer's impression is that the instructor's ability to deliver the complex material is enhanced by the new method.

The PDP - 11 with its limited capacity and the Decwriter terminals with their printing limitations enhance, through symbolism, the realism of the illustrations. In this project, the computer serves to replicate real world conditions and processes. It also provides a mechanism to portray complex information in a way that does not maximize the mechanics or tools. There is no impairment of the course conceptual approach, nor competition with essential sections in the survey. By eliminating the procedural problems and improving the learning processes, understanding of essential real world economic processes is facilitated. The processes that are performed by the computer simulate the role and decision making by the planners. The student is involved in a relationship with the machine that simulates the role of the sector manager and the planners. The nature of the program and substantive information and the actions of the machinery add appropriate simulation of the real world.

## Appendix I. Illustrative Print Out: Input-Output Construction and Illustrations

```
REMEMBER THE DIRECT COEFFICIENTS INDICATE THE FIRST ROUND EFFECTS OF OUT-
PUT CHANGES ONLY. TOTAL ADDITION TO OUTPUT RESULTING FROM ADDITIONAL DE-
MAND CHANGES IS NOT AVAILABLE FROM THE NUMBERS IN TABLE 9. INDIRECT COEF-
FICIENTS ARE NEEDED AS WELL. TOTAL REQUIREMENTS, BOTH DIRECT AND INDIRECT
ARE CALCULATED BY TAKING THE DIFFERENCE BETWEEN AN IDENTITY MATRIX AND THE
INPUT COEFFICIENT MATRIX. TABLE 10 SHOWS THE RESULTS OF THIS PROCESS AND
ILLUSTRATES THE TOTAL OUTPUT REQUIREMENTS.


TABLE 10     DIRECT AND INDIRECT (TOTAL INPUT) COEFFICIENTS

INPUT  // OUTPUT USER
-------//-----------------------------------------------------------------
       //               P R O C E S S I N G
       //-----------------------------------------------------------------
       //   A        B        C        D         E  ....       Z
-------//-----------------------------------------------------------------
  A    // 1.24087  .03356   .04382   .53042    .01979....    .01233
  B    //  .17385 1.62220   .01580   .10961    .01288....    .00934
  C    //  .08391  .02979  1.05695   .03953    .00280....    .00190
  D    //  .00781  .00220   .00531  1.01994    .00612....    .00347
  .    //
  .    //
  .    //
  Z    //  .20293  .13441   .31237   .16622    .33398....   1.04532


TABLE 10 SHOWS THE TOTAL PRODUCTION DIRECTLY AND INDIRECTLY REQUIRED FROM
THE INDUSTRY AT THE TOP FOR EACH DELIVERY OF FINAL GOODS DEMAND BY THE
INDUSTRY AT THE LEFT. EACH TIME "A" SELLS AN ADDITIONAL UNITS WORTH OF GOODS
TO HOUSEHOLDS, GOVERNMENT OR OTHER COMPONENTS OF FINAL DEMAND, "B"'S  OUTPUT
GOES UP .03356, "C"'S  OUTPUT INCREASES .04382 AND SO ON ACROSS THE FIRST ROW.
ALL OTHER ROWS IN THE TABLE ARE READ THE SAME WAY.

EACH COLUMN SHOWS THE OUTPUT REQUIRED FOR A SINGLE SECTOR (DIRECTLY AND IN-
DIRECTLY) TO SUPPORT ONE UNIT OF DELIVERY OF FINAL DEMAND BY EACH IF THE PRO-
CESSING SECTORS. THUS TABLE 10 IS A GENERAL SOLUTION TO THE INTERNAL CONSIS-
TENCY PROBLEM, AND IS ONE MEANS OF ILLUSTRATING ECONOMIC INTERDEPENDENCE. A
TABLE WITH SIMILAR COEFFICIENTS CAN BE USED TO SHOW HOW A CHANGE IN DEMAND
FOR THE OUTPUT OF ONE SECTOR STIMULATES PRODUCTION IN OTHER SECTORS.
```

## Appendix II. Illustrative Input-Output Use in Interim Plan Changes

```
AN ANNOUNCEMENT IS MADE BY THE ECONOMIC COUNCIL THAT THE NEWLY IMPORTED CARS
WILL BE LESS EFFICIENT BECAUSE OF WESTERN BLOC POLLUTION CONTROL EQUIPMENT AND
AN INCREASE WILL BE NEEDED IN DOMESTIC OIL PRODUCTION. SO OIL OUTPUT WILL BE
CHANGED IN 1978, 1979 AND 1980.

THE CPA IS ASKED BY THE EC TO PROJECT THE IMPACT OF THE INCREASED OIL PRODUC-
TION FOR THE THREE YEAR PERIOD 1978-1980 WITH AN INCREASE OF 2% PER YEAR IN


SECTOR  6, OIL.


                                1978               1979               1980
     OIL                      10563.5            10766.6            10969.8
     FERROUS METALS           12708.8            12892.1            13105.9
     AUTOMOBILES               5552.8             5638.6             5713.7
     MEAT + POULTRY           24619.2            25137.4            25608.6


*        *        *        *        *        *        *        *        *        *        *
*                                                                                          *
*  TO: SECTORS  1,6,11,23                                                                  *
*  FROM: CPA                                                                               *
*  RE: REVISED GVO GOALS                                                                   *
*                                                                                          *
*       INCREASED OIL PRODUCTION HAS GENERATED NEW GVO TARGETS FOR                         *
*       YOUR INDUSTRIES STOP ENCLOSED CHART SHOWS OLD 1980 GVO TARGETS                     *
*       AND THE NEW, LOWER GVO PRODUCTION TARGETS STOP PLEASE CONFIRM                      *
*       NEW TARGETS STOP                                                                   *
*                                                                                          *
*                                  1980          REV 1980                                  *
*            OIL                  10665.1          10969.8                                 *
*            FERROUS METALS       13329.8          13105.9                                 *
*            AUTOMOBILES           5740.6           5713.7                                 *
*            MEAT + POULTRY       25679.3          25608.6                                 *
*                                                                                          *
*        *        *        *        *        *        *        *        *        *        *
```

## Appendix III. Illustrative Telegram Communication

```
*      *      *      *      *      *      *      *      *      *      *      *      *      *
*                                                                                          *
*  TO: ECONOMIC COUNCIL                                                                    *
*  FROM: CPA                                                                               *
*  RE: PRODUCTION TARGET CONFIRMATION                                                      *
*                                                                                          *
*                                                                                          *
*        ALL SECTORS EXCEPT  9 CONFIRMED FIVE-YEAR PRODUCTION TAR-                         *
*        GETS STOP EXPECTED SHORTFALL IN  9 ADVERSELY AFFECTS TAR-                         *
*        GETS FOR  1,11 AND 23 STOP REQUEST ADJUSTMENT TO BE MADE                          *
*        IN TARGETS FOR THOSE SECTORS STOP                                                 *
*                                                                                          *
*      *      *      *      *      *      *      *      *      *      *      *      *      *


                     1976         1977         1978         1979         1980
FERROUS METALS      12385        12495        12603        12756        12847
TOOLS + DIES          566          570          574          578          582
AUTOS                5441         5513         5589         5664         5739
MEAT + POULTRY      23989        24439        24663        25007        25361
```

```
*     *     *     *     *     *     *     *     *     *     *     *     *      *
*                                                                            *
*  TO: CPA                                                                   *
*  FROM: EC                                                                  *
*  RE: TARGET ADJUSTMENT                                                     *
*                                                                            *
*          SHORTFALL IN SECTOR  9 CAN NOT BE PERMITTED STOP ORIGINAL         *
*          PRODUCTION TARGETS STILL VALID STOP                               *
*                                                                            *
*     *     *     *     *     *     *     *     *     *     *     *     *      *

*     *     *     *     *     *     *     *     *     *     *     *     *      *
*                                                                            *
*  TO: SECTOR  9 MANAGERS                                                    *
*  FROM: CPA                                                                 *
*  RE: TARGET SHORTFALL                                                      *
*                                                                            *
*          YOU WILL BE REPLACED TOMORROW 9 A.M. BY A SET OF CPA              *
*          TRAINED MANAGERS WHO WILL OVERCOME THE PROBLEMS YOU               *
*          SEEM TO FORESEE STOP YOU HAVE BEEN REASSIGNED TO THE
*          CPA TO AID THE PLANNING GROUP STOP                                *
*                                                                            *
*     *     *     *     *     *     *     *     *     *     *     *     *      *
```

# Use of a Deterministic Macroeconomic Computer Model as a Teaching Aid in Economic Statistics

**John R. Tedford**
*School of Management*
*Clarkson College of Technology*
*Potsdam, New York 13676*
*(315) 268-6434*

## Introduction

The literature about developments in economics education over the past decade reveals a renewed interest in the never-ending search for new and better ways of improving the undergraduate learning process and the teaching of economics. Major emphasis, of late, has been directed at questions of methodology and at techniques which may be used in place of or as complementary aids to the traditional lecture approach. As one might expect, these developments are similar to those occurring in other subject-matter areas. Presumably improvements in the learning techniques for one area will have interdisciplinary benefits. In this sense it appears that some of Professor Bork's [1] general comments and hypotheses about the use of computers are as meaningful for the teaching of economics as they are for physics. He argues that the test of all learning is with students and the value of teaching materials, computer or otherwise, depends upon whether or not they lead to some type of learning for some students. In effect Bork, much like Kelley [5], suggests that the learning process is an individualistic or micro process. If we subscribe to this position and expect to be successful in improving upon our traditional teaching activities, a variety of techniques and instruments designed to meet the individual student's needs will be required.

The field of economics has seen an increasing number of new techniques developed and used over the past several years. Particular attention has been directed at the use of computerized macroeconomic models in the principle courses. Such models were often designed and used in the hope that students would obtain a better understanding about modern macroeconomic concepts and how they relate to questions of public policy. For example, Attiyeh [5] claims that students often cannot deal successfully with practical policy problems because they misunderstand the connection between economic analysis and its application to the real world. He states, "One way to counter this misunderstanding is to let students make concrete policy decisions on their own, using the Macroeconomic Policy game. . . .Consequently, (students) learn to carry out economic analysis and come to view it, not as an irrelevant logical exercise, but rather as the useful policy tool that it is."

In a discussion paper Bower [2] raises some important questions about the use of macroeconomic computer games as teaching experiments. He argues that while students have fun in playing computerized games, they may not learn much if not much is asked of them. He states, "They have fun and learn something when they set values for policy variables in an effort to achieve output, employment, and price targets. But wouldn't they learn a great deal more if they were forced to formulate their analysis, make their model explicit and provide numerical estimates of relevant relationships. . . With an explicit model and a derived table of multipliers, students should learn more from analysis than they ever could from errors such as confusing tax-income causality."

No doubt an experiment designed to have students obtain estimates of relevant structural parameters for an explicit model and to make policy decisions based upon them would provide a more complete and effective learning experience. One suspects, however, that such an experiment would be meaningful only for those students who had prior knowledge about model building and about statistical estimation techniques. These prerequisites are much greater than one could normally expect to find in undergraduate students taking a beginning macroeconomic principles course. They are topics one would normally expect to find covered in courses in economic statistics or econometrics. Presumably use of the above type of experiment would be more appropriate in such a course.

## The Problem

The purpose of this paper is to describe how a simple deterministic macroeconomic computer model was used as an experiment in a junior-senior level economic statistics course. The objective of the experiment was to demonstrate how and why some of the errors commonly arise when statistical estimation techniques are applied to economic relationships in empirical problematic situations. The computer model was treated as the true universe or real-world system and each student was required to use it to generate his own sample data. The basic hypothesis underlying the experiment was—if students really understood (1) the underlying concepts and methods of statistical estimation of economic relationships, and (2) the reasons why the basic assumptions underlying these methods may be violated when applying them empirically, then they would make decisions in such a way that the resulting sample data and parameter estimates would be free of all estimation errors other than sample errors.

A deterministic macroeconomic computer model was designed and used to broaden the student's problem, in hopes that it would provide a better learning experience. The estimation of parameters for economic relationships by multiple regression or other techniques involves use of two models—one economic and the other probabilistic or statistical. Many of the empirical estimation errors arise because of differences or inconsistencies in or between these two models. That is, in applying statistical estimation techniques we generally assume that the specifications for the economic model and the available data satisfy the requirements and assumptions of the underlying statistical model. All too often some aspect of these models, as traditionally formulated and applied, will be violated. Due to differences between the hypothesized and actual conditions, a number of problems may arise. Accordingly, if the students were to obtain meaningful parameter estimates in the experimental situation, they would have to consider the relationships between the economic and statistical models. Use of a deterministic model as the true information generating mechanism compounded the problem because the students had to provide the random or stochastic elements consistent with the statistical model. That is, the students were expected to make decisions in a manner which explicitly generated data possessing the prerequisite random characteristics, or they would encounter such estimation errors as multicollinearity, autocorrelation, heteroskedasticity, or specification errors of various types, etc.

## The Experiment

The overall class assignment may be looked upon as a set of estimation experiments. That is, each student's use of the computer model, as the true population mechanism, to generate sample data and to obtain parameter estimates constitutes an experiment. By varying the types and amounts of information made available to students about the model's specifications, initial conditions and/or computer print-out data, one can alter the structure of the experiments. As mentioned earlier, many of the estimation errors arise because the traditionally formulated economic and statistical models are applied in a world of less-than-complete or perfect knowledge. When the information and specifications provided by the instructor are not complete, the student is forced to provide his own a priori hypothesis and specifications which may or may not be consistent with the true model's structural characteristics. Likewise, the student has to evaluate and analyze his computer-generated sample data in much the same way as he has to analyze real-world data. Either situation can be a source of estimation errors. One of the major advantages of this type of experiment is that each student's statistical estimates can be compared with the true parameter values. When discrepancies exist, the class can be called upon to evaluate each experiment's results and to provide an explanation of the types of errors and their possible causes.

The above framework was used with all students in the economic statistics class. The macroeconomic computer model was a revised version of one developed by Professor Attiyeh [5]. Each student was assigned an independent economy having the same parameter values and initial conditions. The instructions included: (1) a description of all of the relationships comprising the model, (2) definitions for each variable, (3) values for initial conditions of the system, (4) parameter values for all relationships other than the unemployment and the rate of inflation functions, and (5) the assigned problem or task. Since each student had prior experience and familiarity with statistical estimating programs on the computer, no information of this type was needed.

The specific assignment consisted of two parts. Each student was told to generate a sample of any size he wished but to use a sample of only twenty observations to obtain estimates for the parameters of the Phillips curve. Each student was also asked to keep a record of the decision rules and procedures he used in generating data and in obtaining parameter estimates.

The class discussed the concept of the Phillips curve and its use to reflect the tradeoff between unemployment and inflation. No information was provided about the specific mathematical nature of the Phillips curve in the true model other than that provided in the model's description. Specifications for the Phillips curve may be obtained from the unemployment and rate of inflation relationships as given in Appendix A. Manipulation of these relationships indicates that the Phillips curve is given by three sets of relationships depending upon the state of the economy.

The first segment of the Phillips curve applies when the economy is running with inflationary pressures, i.e., equilibrium NNP > full employment NNP. In this case, unemployment (U) will equal a minimum value of 3 percent. Inflation, on the other hand, becomes larger and larger as the difference between the equilibrium value of NNP and the full employment value of NNP gets greater due to equation 16 in Appendix A. That is, the Phillips curve is given as

(1) $F = 306 - 700\,(Y'/Q) + 400\,(Y'/Q)^2$
when $U = 3$ and $Y' > Q$
*where* $Y'$ is equilibrium *NNP* and $Q$ is full employment *NNP*.

If, however, the economy experiences deflationary pressures, i.e., equilibrium *NNP* < full employment *NNP*, then unemployment will vary and be greater than 3 percent. The rate of inflation is generated by one of two relationships, equations 15 or 16, depending upon the ratio that equilibrium *NNP* is to the full employment value of *NNP*. Upon examining these relationships, we find that when unemployment $U$ reaches its minimum value of 3 percent, the rate of inflation $F$ approaches a value of 6 percent. On the other hand, $F$ is zero when U is 5.8 percent and $F$ takes on a negative value when $U$ is greater than 5.8. When unemployment takes on values from 3 to 6 percent, the Phillips curve is obtained by manipulation equations 14 and 16 which yields

(2) $F = 21.28 - 6.62\,U + .51\,U^2$ for
$3 \leq U < 6$.

Third, when unemployment is equal to or greater than 6 percent, the Phillips curve is linear because both equations 14 and 15 are linear. That is, the third segment of the Phillips curve is given as

(3) $F = -.069 - .0018\,U$ for $U \geq 6$.

The computer model was run on an IBM 360/44 under a batch mode. The students received instructions about the procedure for making decisions. They were allowed one week as a trial period to familiarize themselves and to minimize procedural errors. Few restrictions were placed upon the students who could use any combination of three decision variables to manipulate their economy over time. Each could control movements in the equilibrium values of NNP over time by changing (1) government spending, (2) the rate of taxes, and/or (3) the money supply. The students were told several times that the macroeconomic computer model was deterministic, in hopes that they would explicitly consider and build a random process in their sample data when making their decisions.

### Results and Conclusions

Few of the students were successful in obtaining perfect statistical estimates as most results possessed estimation errors similar to those one finds in empirical analysis. In this respect, the class experiment might be judged as having been successful.

Only the results for six of the students' experiments will be described. This is not a serious limitation, however, because the excluded results were basically similar to those which will be described. To simplify discussion, the six students' experiments have arbitrarily been designated by use of the first six Roman numerals.

The decision rule or procedure underlying each of the six students' sequence of decisions (i.e., each economy's time path for the decision variables) were as follows: (1) In Economy I no changes were made in any of the decision variables. That is $\Delta G$, $\Delta TX$ and $\Delta MS$ were each set equal to zero during each quarterly time period. (2) Government spending was the only decision variable used in Economy II. The procedure was to make the change in government spending equal to any amount necessary to attain an unemployment rate of 4 to 5 percent in each time period. (3) The time path for the decision variables utilized in Economy III was similar to that in Economy II. An attempt was made to attain a zero level of inflation in each time period. (4) In Economy IV, both government spending and the money supply were changed in a random manner. Apparently the variance of the random process was not large because the changes made in the decision variables from period to period were small in value. (5) Government spending and the money supply were also altered in Economy V. The money supply was varied in a random manner but government spending was changed by an amount which the student expected would make the equilibrium value of *NNP* equal to 97 percent of full employment *NNP*. (6) In Economy VI the student randomly selected values for the independent variable $Y/Q$ contained in unemployment and inflation equations in the model (Appendix A). The student only used government spending as a decision variable which he changed by an amount expected to yield the randomly generated value of $Y/Q$ for each quarter.

Sample data for the variables $U$ and $F$ for each economy are given in Appendix B. Casual examination of these data suggests that the students assumed that the correct specification for the Phillips curve was but one relationship for all possible values of $U$. Because of this, it appears that specification errors might present some estimation problems in Economies I, IV, and VI. The data also suggest that problems of multicollinearity might be present in Economies II and III. Positive autocorrelation might be found in the statistical estimates for all economies, possibly with the exception of Economy VI.

Since no restrictions were placed upon the specific mathematical functions that the students could postulate, many types were specified and estimated. The estimation results are summarized in Table 1 for the linear and quadratic models as these were the only types of functions commonly used. Even though these results are partial and incomplete since they show the findings for only a part of the class, they do offer some interesting conclusions. They are interesting both with respect to their statistical characteristics and the objective of this paper.

**Table 1. Students' Statistical Estimates for the Parameters of the Phillips Curve**[1]

I. Linear Model: $F = a + b\,U + e$

| *Economy* | | | | | |
|---|---|---|---|---|---|
| *I* | $F = 1.003$ | $- .178\,U$<br>(− 625.) | | $R^2 = 1.$ | $D = 1.468$ |
| II | $F = 8.591$ | $- 1.535\,U$<br>(− 75.6) | | $R^2 = .9969$ | $D = .853$ |
| III | $F = 2.795$ | $- .472\,U$<br>(7.6) | | $R^2 = .76$ | $D = 2.09$ |
| IV | $F = 40.67$ | $- 7.08\,U$<br>(− 3.9) | | $R^2 = .46$ | $D = 0.50$ |
| V | $F = 5.87$ | $- .84\,U$<br>(− 9.2) | | $R^2 = .83$ | $D = 1.29$ |
| VI | $F = 74.5$ | $- 11.46\,U$<br>(− 1.7) | | $R^2 = .14$ | $D = .71$ |

II. Quadratic Model:[2] $F = a + b\,U + c\,U^2 + e$

| *Economy* | | | | | |
|---|---|---|---|---|---|
| *I* | $F = 1.026$ | $- .18\,U$<br>(− 73.9) | $+ .0002\,U^2$<br>(1.88) | $R^2 = 1.$ | $D = 1.887$ |
| IV | $F = 101.9$ | $- 35.58\,U$<br>(− 2.6) | $+ 3.1\,U^2$<br>(2.12) | $R^2 = .57$ | $D = .51$ |
| V | $F = 18.0$ | $- 5.26\,U$<br>(− 19.1) | $+ .37\,U^2$<br>(16.1) | $R^2 = .989$ | $D = 2.72$ |
| VI | $F = 201.7$ | $- 69.3\,U$<br>(− 1.02) | $+ 5.76\,U^2$<br>(.85) | $R^2 = .17$ | $D = .79$ |

---

1. The values in the parentheses under the coefficients in each equation are the values of students' $t$ statistic.

2. No results were available for this model for Economies II or III and no explanation was provided.

Upon analyzing and comparing the statistical estimates and associated statistics for the linear models, one finds that the overall findings for Economy I yield the best results. This is what one might expect to obtain because all of the values for $U$ in Economy I are greater than six, quite consistent with the linear specification for the third segment of the True Phillips curve. The observations for all other samples are inconsistent with the specifications for this segment of the Phillips curve. Accordingly, the statistical estimates for each of the economies other than I appear to have some type of estimation error. For example, while the parameter estimates for Economy III provide the next best set of results, they overestimate the true parameter values and the linear estimating model only explains about three-fourths of the error in the dependent variable $F$.

Statistical estimates for the quadratic models also vary considerably. The best estimates were obtained for Economy V. The sample data for this economy, while not perfect, are consistent with specifications for the second segment of the True Phillips curve. That is, if one were interested in policy decision over the relevant range of values of $U$ between 3 and 6 percent, Economy V's estimates would provide reasonable approximations. This appears to be so even though the sample estimates underestimate the true parameter values and the test for negative autocorrelation is inconclusive. On the other hand, while the quadratic model explains all of the variations in inflation for Economy I, it is not significantly better than the linear model because the coefficient associated with the $U^2$ term is not significantly different from zero. The overall results for Economies IV and VI are not at all reasonable because of positive autocorrelation and other errors.

The above brief comments illustrate the nature of the discussion that took place in the classroom after the students had handed in their respective findings. Upon receiving the students' findings, the true specifications for the Phillips curve were distributed. The students were then asked to describe the nature of the statistical estimation errors present in the findings and to explain why each error may have occurred. The students' analyses and explanations were not as rich nor as complete as had been hoped. Nevertheless, the overall results supported the contention that macroeconomic computer models can be effectively used in economics courses to teach about concepts and topics other than public policy.

All of the students appeared to enjoy knowing the true parameter values and being able to compare their results with the true values as well as the findings of others. They were all surprised at the variability in their results. While most of the students indicated that they understood the objective of the assignment, apparently several did not fully understand the complexity of their individual assignments. It was apparent in the class discussion that some of the students did not understand that when one uses a deterministic computer model, estimation errors are likely to arise if one does not explicitly look upon the use of the model as an information-generating mechanism and so use it to generate data that are of a random nature. The author believes that treatment of this aspect of the experiment was a major weakness.

Another apparent weakness of the experiment was that the design was far from perfect as there are several ways in which the class's experience could be improved upon. The problem assignment could have been more structured than it was. Much more specific information could have been given to the students and tests could have been used to determine if the students understood the problem and procedural aspects involved in its solution. Nevertheless, the author believes use of the computer model was effective and with appropriate modifications it will be even more effective as a teaching aid in the future.

## References

1. Bork, A., "The Computer in a Responsive Learning Environment—Let a Thousand Flowers Bloom." *Proceedings* of 2nd CCUC, Dartmouth College, June 1971.
2. Bower, R. S., "Discussion of Dolbear, Attiyeh and Brainard's paper—A Simulation Policy Game for Teaching Economics." *American Economic Review*, LVIII, No. 2, May 1968.
3. Kmenta, J. *Elements of Econometrics*. The Macmillan Company, New York, New York, 1971.
4. Lumsden, K. G. (Editor). *New Developments in the Teaching of Economics*. Prentice-Hall, Englewood Cliffs, N.J., 1976.
5. ______________. *Recent Research in Economics Education*. Prentice-Hall, Englewood Cliffs, N.J., 1970.

## APPENDIX A

### The Macroeconomic Model and Definitions for Variables

*The Model*

I. Potential full-employment *NNP*

(1) $A_t = (1 - a_1) A_{t-1}$

(2) $L_t = (1 - a_2) L_{t-1}$

(3) $K_t = K_{t-1} + a_3 I_{t-1}$

(4) $Q_t = A_t \cdot L_t^{a_4} \cdot K_t^{a_5}$

II. Commodity and money markets

(5) $C_t = b_1 + b_2 \cdot YD_t$

(6) $I_t = b_3 + b_4 (Q_t \cdot R_{t-1}) + b_5 Y_t + Z_{t-1}$

(7) $BS_t = b_6 Y_t$

(8) $G_t = G_{t-1} + \Delta G$

(9) $YD_t = Y_t - TX_t - BS_t$

(10) $TX_t = b_7 + b_8 Y_t$

(11) $MS_t = MS_{t-1} + \Delta MS$

(12) $R_t = b_9 + b_{10} (MS_t / P_{t-1}) + b_{11} Y_t$

(13) $Y_t = C_t + I_t + G_t$

III. Other relationships

(14) $U_t = 3 + 28(1 - Y_t / Q_t)$

(15) If $Y_t/Q_t \quad .89$
$F_t = -.08 + .05(Y_t/Q_t - .89)$

(16) If $Y_t/Q_t \quad .89$
$F_t = 306 - 700(Y_t/Q_t) + 400(Y_t/Q_t)^2$

(17) $Z_t = Z(Q_t, Y_t)$

(18) $P_t = P(F_t)$

(19) $W = W(C, I, G, U, F)$

*Definitions*

*A* — state of technology
*L* — labor force
*K* — capital stock
*Q* — full employment NNP
*C* — desired consumer spending
*I* — desired investment spending
*BS* — business savings
*G* — government spending
*TX* — government taxes
*Y* — NNP
*YD* — disposable income
*MS* — money supply
*R* — interest rate
*U* — percent unemployment
*F* — rate of inflation
*P* — price level
*Z* — inflationary inventory disinvestment factor
*W* — welfare index

## APPENDIX B

### Sample Data Generated by Students for Variables *U* and *F*

| *Economy I* | | *Economy II* | | *Economy III* | |
|---|---|---|---|---|---|
| % Unemployment | Rate of Inflation | % Unemployment | Rate of Inflation | % Unemployment | Rate of Inflation |
| *U* | *F* | *U* | *F* | *U* | *F* |
| 6.47 | − .15 | 4.79 | 1.25 | 6.23 | −.11 |
| 6.87 | − .22 | 4.76 | 1.30 | 6.07 | −.15 |
| 7.44 | − .32 | 4.78 | 1.25 | 6.6 | −.17 |
| 7.94 | − .41 | 4.96 | .96 | 6.57 | −.17 |
| 8.44 | − .50 | 4.67 | 1.45 | 5.58 | .19 |
| 8.92 | − .59 | 4.76 | 1.30 | 5.86 | −.04 |
| 9.38 | − .67 | 4.76 | 1.29 | 5.89 | −.06 |
| 9.83 | − .75 | 4.78 | 1.25 | 6.0 | −.12 |
| 10.26 | − .83 | 4.81 | 1.20 | 5.32 | .46 |
| 10.68 | − .9 | 4.84 | 1.16 | 5.67 | .10 |
| 11.08 | − .97 | 4.84 | 1.16 | 5.74 | .04 |
| 11.47 | −1.04 | 4.87 | 1.11 | 5.98 | −.11 |
| 11.85 | −1.11 | 4.91 | 1.04 | 5.37 | .40 |
| 12.22 | −1.18 | 4.97 | .94 | 5.73 | .05 |
| 12.58 | −1.24 | 5.00 | .90 | 5.82 | −.01 |
| 12.92 | −1.30 | 5.05 | .82 | 6.07 | −.16 |
| 13.26 | −1.36 | 5.12 | 72 | 5.54 | .22 |
| 13.59 | −1.42 | 5.2 | .61 | 5.89 | −.06 |
| 13.91 | −1.48 | 5.26 | .53 | 6.0 | −.12 |
| 14.22 | −1.53 | 5.34 | .43 | 6.26 | −.11 |

## Appendix B *(con't.)*

| Economy IV | | Economy V | | Economy VI | |
|---|---|---|---|---|---|
| % Unemployment | Rate of Inflation | % Unemployment | Rate of Inflation | % Unemployment | Rate of Inflation |
| *U* | *F* | *U* | *F* | *U* | *F* |
| 5.39 | .38 | 3.84 | 3.36 | 6.92 | −.23 |
| 5.58 | .18 | 8.26 | −.47 | 3.34 | 4.85 |
| 6.7 | − .19 | 7.34 | −.31 | 6.93 | −.23 |
| 6.14 | − .09 | 4.12 | 2.64 | 5.73 | .05 |
| 7.01 | − .25 | 4.36 | 2.07 | 3.15 | 5.48 |
| 4.57 | 1.65 | 4.95 | .98 | 7.01 | − .25 |
| 5.24 | .56 | 4.62 | 1.54 | 5.76 | .03 |
| 4.44 | 1.92 | 4.68 | 1.44 | 3.00 | 6.72 |
| 5.74 | .05 | 4.00 | 2.95 | 6.65 | −.18 |
| 5.13 | .71 | 4.70 | 1.40 | 5.75 | .04 |
| 5.76 | .03 | 3.85 | 3.3 | 4.34 | 2.13 |
| 4.64 | 1.51 | 4.08 | 2.75 | 3.00 | 7.98 |
| 4.86 | 1.12 | 4.09 | 2.72 | 3.99 | 2.96 |
| 3.00 | 6.0 | 4.12 | 2.65 | 7.82 | −.39 |
| 3.41 | 4.62 | 4.19 | 2.47 | 6.69 | −.19 |
| 3.00 | 8.92 | 4.34 | 2.14 | 3.00 | 7.87 |
| 3.00 | 17.82 | 4.32 | 2.17 | 3.00 | 8.49 |
| 3.00 | 24.57 | 4.47 | 1.86 | 3.00 | 23.08 |
| 3.00 | 40.64 | 4.59 | 1.61 | 3.00 | 95.65 |
| 3.00 | 47.71 | 4.60 | 1.58 | 3.00 | 236.71 |

# DISCOVER: A Computerized Careers Curriculum

**Jack R. Rayman**
*Western Maryland College*
*Westminster, Maryland 21157*
*(301) 848-1113*

## Introduction

Critics of post-secondary education and particularly liberal arts education have become increasingly vocal in recent years. Caroline Bird's book, ***The Case Against College***, calls college the "dumbest investment of all." Colleges themselves seem to be less than secure as evidenced by their recent prime-time media blitz in an attempt to convince the public of their slogan, "College—it's worth it." It is this author's premise that much of the "irrelevance" ascribed to post-secondary education results from the inability and unwillingness of the liberal arts faculty to acknowledge that most students will some day be forced to "earn a living" so that they can afford the luxury of "living a life." The "good life" as liberal arts advocates envision it is an option to most people only if they are also fortunate enough to have "the good job." As Blanche D. Blank put it in her ***AAUP Bulletin*** article entitled, "Degrees: Who Needs Them?,". . . there is far more evidence that work can make college meaningful, then that college can make work meaningful." A recognition of this fact may eventually lead to the so-called "infusion of career education" into the curriculum of our educational institutions as called for by Kenneth Hoyt and others in the U.S. Office of Education (1975).

In the meantime, however, students continue to hold unrealistic expectations about the value of a college education as occupational preparation. Most students have poorly specified career objectives, a limited understanding of the occupational world, limited access to occupational information and placement facilities, and poorly developed job search skills. A survey of university seniors regarding the frequency of student needs for student personnel services conducted by Carney and Barak (1976) at Ohio State University found that the number-one area of concern (expressed by 37 percent of those surveyed) was choice of major or career. Clearly, college and university students feel the need for improved career services and delivery systems. McGrath (1974), along with numerous administrators in higher education, suggests that, in fact,the success of many institutions in retaining and recruiting students will lie in their willingness to provide services for undergraduates that relate careers, values, and general education.

Perhaps the best solution to this problem would be the integration of career development theory and relevant career information into the curriculum of all college major programs. Unfortunately such an approach meets great resistance from many faculty members to whom the term "career education" is an anathema. Liberal arts faculty members are particularly sensitive to any program which might alter the purity of their discipline. To them the idea of injecting career information into the curriculum conjures up thoughts of the ivy covered liberal arts campus being overrun by legions of wrench toting auto mechanics. Because of this resistance, most liberal arts campuses have resorted to developing separate career courses (usually non-credit) in an attempt to meet the student demand for greater career understanding and awareness. Such courses have met with only limited success.

## Computers and a Career Curriculum

In the past decade a number of concerned social scientists have been studying career development with an eye to developing treatments and interventions which may facilitate career development outside the traditional classroom situation. One highly innovative approach has been the use of the computer to deliver career information and services via cathode ray tube terminals. Recently, the DISCOVER Foundation (located on the Campus of Western Maryland College) has developed a comprehensive computerized career guidance system for grades seven through twelve which not only provides career information, but also performs such tasks as teaching students decision-making skills and helping them to clarify their values.

Through the financial support of the Exxon Education Foundation, this author is now directing the adaptation of the DISCOVER Computerized Career Guidance System for use at the college level. This college-level system will provide a computerized careers course which will allow college students to greatly enhance the value of their education by increasing its relevance to the occupational world. The computer has a number of capabilities which make it a particularly valuable tool for the purposes of delivering career information and services.

1. It can store, search, retrieve, relate, and update large data files with ease and accuracy.
2. It can interrelate computer-stored data about a user with data about schools and occupations in a way that creates new information for the user.
3. It is capable of simulating structured interviews with a high degree of "personalization" through the use of cathode ray tube terminals with light pens.
4. It is capable of providing individualized attention to many users simultaneously for many hours per day.
5. It is capable of monitoring a user's progress in career decision making and of providing appropriate feedback, clarification, or review.
6. It has the potential to provide high quality career service without being perceived as a threat by the faculty.

Further evidence in support of using the computer in the delivery of career services stems from the evaluation of its effectiveness in currently operational systems at the high school and junior high levels (Bowlsbey, Rayman & Bryson, 1976; Pyle & Stripling, 1976; Maola, 1974; Chapman, Norris & Katz, 1973; Harris, 1972; Myers, et. al., 1971; Melhus, 1971; Price, 1971; Impelleteri, 1968). A summary of the evaluation of existing computer-based systems indicates that:

1. They are enthusiastically accepted and easily used by students.
2. They cause an increase in conversation about career planning.
3. They cause measured increases in vocational maturity, occupational exploration, occupational knowledge, self-knowledge, and specification of career plans.
4. Students prefer to get career information from a computer rather than from books, files, and other traditional sources.
5. Some career guidance functions can be done as well by the computer as by counselors, at a lower per-hour cost.
6. The computer is perceived by many users as being a less threatening purveyor of career information than a counselor.

Based on the capabilities and evidence outlined above, DISCOVER—a computerized career development system—appears to be an ideal means of delivering a "careers curriculum" to college students.

## Content of the DISCOVER System

DISCOVER is a systematic career development program designed to enhance normal career development for college-age students. The study of career development theory makes it clear that a career development system, or "careers curriculum," should include at least the following components: (1) self-information, including values, interests and competencies, (2) exploration of occupations in a systematic way, (3) teaching and low-risk practice of career decision making, (4) relationship of self-information to occupational alternatives, and (5) informational assistance with the implementation of career choice.

DISCOVER provides assistance in all of these areas through twelve fifty-minute modules of interactive content. Data about the user—including name, course, and grades; test information; extracurricular activities; and work experience—are stored in the system for use in comparing occupational or training requirements with the user's past experience. Some tools and techniques developed and copyrighted by other guidance professionals are used within DISCOVER by permission, and in some cases with royalty agreements. Examples of these are Holland's Self-Directed Search (1972) and Super's Career Decision Tree (1975). Seven large data files—occupations, four-year colleges, two-year colleges, technical and specialized schools, apprenticeships, military training programs, and graduate and professional schools—support the student's search for occupational and educational alternatives. In summary, DISCOVER is a combination of originally developed interactive dialogue and simulations, of instruments and tools used by permission of other developers, and of originally developed and borrowed data files.

The "main line" modules of the system are graphically depicted in Figure 1. The titles and numbers of each module correspond to the brief descriptions provided in a later portion of this paper. The position of each module on the arrow-shaped diagram is based on a developmental approach to career guidance. Super's Developmental Stages (1963) and Tiedeman & O'Hara's Decision-making Paradigm (1963) are plotted at the bottom of the diagram to depict the association which exists between the content of each module and these two developmental theories. In the DISCOVER system a person of low vocational maturity would be advised to begin with Module 1A (at the left end of the arrow) and to proceed in numerical order through Module 8 and its sub-modules (at the tip of the arrow). DISCOVER does, however, accommodate individual differences of interest and vocational maturity by allowing the user to enter the system at any point, to use it in any order, or to receive a suggested sequence through the system by taking an instrument called the Survey of Career Development which is contained in the entry module (an administrative section of the system which orients the user to the system and monitors use). The system contains one administrative module (the entry module), eleven "main line" modules and nine informational modules which are described here.

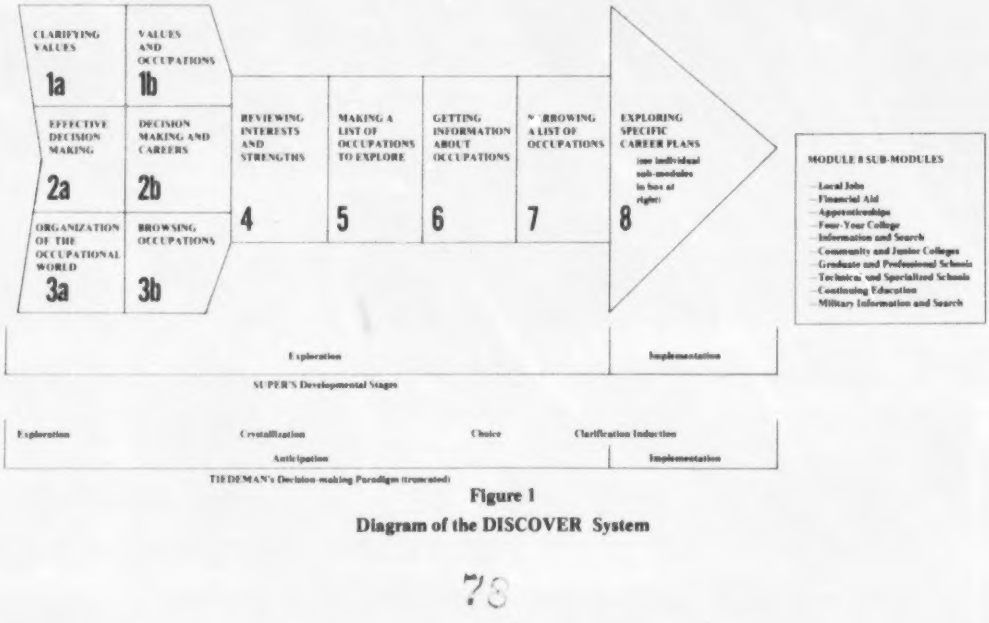

CLARIFYING VALUES
1a
VALUES AND OCCUPATIONS
1b
EFFECTIVE DECISION MAKING
2a
DECISION MAKING AND CAREERS
2b
ORGANIZATION OF THE OCCUPATIONAL WORLD
3a
BROWSING OCCUPATIONS
3b
REVIEWING INTERESTS AND STRENGTHS
4
MAKING A LIST OF OCCUPATIONS TO EXPLORE
5
GETTING INFORMATION ABOUT OCCUPATIONS
6
NARROWING A LIST OF OCCUPATIONS
7
EXPLORING SPECIFIC CAREER PLANS
(see individual sub-modules in box at right)
8
MODULE 8 SUB-MODULES
—Local Jobs
—Financial Aid
—Apprenticeships
—Four-Year College
—Information and Search
—Community and Junior Colleges
—Graduate and Professional Schools
—Technical and Specialized Schools
—Continuing Education
—Military Information and Search
Exploration
Implementation
SUPER'S Developmental Stages
Exploration
Crystallization
Choice
Clarification Induction
Anticipation
Implementation
TIEDEMAN's Decision-making Paradigm (truncated)

**Figure 1**

**Diagram of the DISCOVER System**

### Entry

The entry module introduces the user to the many special features of the system and teaches him/her how to use the terminal. It monitors each person's use of DISCOVER, recording each entry and exit point to facilitate long-term use and review. Each user has the opportunity to complete an online Survey of Career Development (Rayman & Super, 1975). This survey is scored and the results serve as the computer's guide in suggesting which modules of the system will be most appropriate for a given user.

### Clarifying Values

This module contains a number of experiences which lead the user to think about what a value is, to analyze personal values, and to decide upon actions to implement those values.

### Values and Occupations

This module defines nine specific occupationally-related values as identified by Katz (Chapman, Norris & Katz, 1973). These definitions are followed by a series of exercises to help the user develop a good understanding of the nine occupationally-related values. The student may then rate the personal importance of each of these nine values and ask the computer to search its data file for occupations which can provide the combination and weighting of the values assigned. The list of occupations thus generated is retained for later use in the system.

### Effective Decision Making

This module attempts to teach a planful decision-making process by presenting the steps of the decision-making process in example and flowchart form. The system provides a number of exercises designed to illustrate and provide practice in the decision-making strategies (impulsive, delaying, painful) and assists the user to identify his/her present style of decision making.

### Decision Making and Careers

This module makes use of a "career decision tree" (Super, 1975) as an organizing principle for understanding how decisions affect occupational choice. Since each of the twenty branches of the tree represents a group of occupations, the tree structure is used (1) to show the key decisions which led to entry into a given occupation, (2) to plot a given user's course up the branches of the tree, (3) to simulate the career paths of others, and (4) allow the user the opportunity to "play" his/her own life in a variety of ways by making decisions in this low-risk simulation.

### Organization of the Occupational World

This module presents the world of work by way of two organizing principles; the data-people-things-ideas division (which is the American College Testing Program's refinement of the *Dictionary of Occupational Titles* classification system) and the Holland (1973) system. A number of exercises are presented to give the student practice at using these classification systems; the student's responses are monitored for the purpose of providing more instruction if needed.

### Browsing Occupations

This module makes use of the Holland classification system presented in Module 3A as an organizational structure by which the user can browse the world of work. The module allows him/her to touch any two points of the Holland hexagon and to receive a list of occupations which fall in that particular segment of the circular world of work. The user may select titles from the list and receive a one-display description of the occupation's work setting and work tasks. This module contains descriptions of 320 occupations.

### Reviewing Interests and Strengths

This module is John Holland's *Self-Directed Search* (1972). This instrument is a self-report of the user's career-related interests, experiences, and competencies. The

### *Making a List of Occupations to Explore*

This module provides the user with five alternate ways to make a list of personal vocational options (1) by relationship of occupations to personal work values, (2) by use of the results of the *Self-Directed Search*, (3) by selecting titles from a list of occupations by the terminal, (4) by combining selected occupational characteristics (such as salary level, place of work, level of training, degree of independence, etc.) and/or (5) by relating favorite school subjects to occupations.

### *Getting Information About Occupations*

This module allows the user to get extensive information about the occupations on his/her list. By the voluntary selection of as many as twenty-one questions, the user may receive information about an occupation, its duties, benefits and limitations, educational requirements, future outlook, and additional sources of information. The user may also review his/her own student record (of grades in related courses, related work or course experience, present rank in class, etc.) against the requirements of the occupation or its prerequisite training. The module also provides the capability to compare two occupations by calling in data about both simultaneously. The user leaves this module with a list of occupations in which he/she has serious interest. This list may be a shortened form of the list with which the user entered the module, or it may be a new list which has been generated on the basis of information gathered in this module.

### *Narrowing a List of Occupations*

The user enters this module with the list of occupations from the previous module or with a new one which he/she generates at the beginning of the module. The purpose of this module is to assist the user to narrow the list further so that he/she leaves the module with a first choice occupation in mind and a limited number of others in priority order. This narrowing is assisted by the capability to (1) ask for additional information about any occupations on the list, (2) compare information about two occupations, and (3) analyze the remaining occupations in light of identified work values, desired level of training, and interests and competencies. Finally, the user is asked to remove occupations which are no longer of interest and to put the others in priority order. The user leaves the module with a top-priority selection.

### *Exploring Specific Career Plans*

The user enters this module with one specific occupation in mind at a time. The system identifies for the user all of the possible paths of training to the selected occupation. The user may choose the path of entry which he/she wishes to explore in depth and branch to any of the following sub-modules of the system.

### *Local Jobs*

This module provides an informational section about how to seek a job. It also provides a search strategy for finding a job in the local area with the characteristics desired by the user. Use of the search strategy requires that each local site develop a local job data bank.

### *Financial Aid*

This module provides three capabilities: (1) definition of standard financial aid terms; (2) information about how to obtain financial aid, including an on-line assessment of student financial need (based on a short form of ACT's Pre-planning Financial Statement); and (3) information about available sources of financial aid.

### *Apprenticeships*

This module provides two capabilities: (1) answers to common questions about apprenticeship programs; and (2) search strategies to find apprenticeable occupations, including locating local companies which sponsor apprentices in the user's chosen occupation (requires local data file development).

data, collected via the items on the instrument, give the user a focus for exploration in the world of work. The results of the instrument are interpreted to the user "on-line."

### *Four-year College Information and Search*

This module is a very extensive one which has two divisions: (1) exhaustive information about college admission and selection, and (2) a search strategy. The search strategy draws upon a data file of all four-year colleges and universities in the nation, collected and maintained by the American College Testing program. The strategy allows the user to combine characteristics in the following areas: approximately 200 majors, cost ranges, enrollment ranges, geographic locations, levels of competition for admission, student characteristics (test and class rank, sex, and racial/ethnic membership), type of control, type and size of community, type of institution, type of school year calendar, and a variety of special features. Combination of the student's selected variables produces a list of colleges which have the desired characteristics. Five displays of specific information are available about each college: entrance requirements, general information, characteristics of the freshman class, majors offered, and cost and financial aid.

### *Community and Junior Colleges*

This module provides two capabilities: (1) answers to common questions about community and junior colleges, and (2) search strategies to find a college. Two strategies are provided: search of local colleges only or search of a nationwide file. DISCOVER sites may choose to use only the local search. The DISCOVER-provided college data file contains the data with which to do both searches.

### *Graduate and Professional Schools*

This module provides a search through a national data file of graduate and professional schools making use of characteristics very similar to those used in the four-year college search.

### *Technical and Specialized Schools*

Similar to other modules in this group, this one provides general information about technical and specialized schools and a search strategy to find those which will provide the training needed for entry into a specific occupation. The data file provided by DISCOVER is a national one, containing information about approximately 11,000 schools. The local site may choose to use only a part of this file.

### *Continuing Education*

This program has two sections: (1) general information about adult and continuing education, and (2) instruction on how to find adult-continuing education opportunities in the local area. A local site might choose to enhance this module by adding a local data file of continuing education opportunities.

### *Military Information and Search*

This module provides answers to a large number of typical questions about the military. Additionally, it provides a search strategy to find the specific program within any branch of the military which provides the training needed for entry into a specific civilian occupation. The module contains the data file published in the Department of Defense *Military-Civilian Occupational Source Book* (1975).

## Discussion

In the past decade a great deal of time, energy, and effort have gone into the development of computer-assisted instructional technology for the delivery of courses in the traditional disciplines. The project described in this paper has drawn on this technology to deliver a not-so traditional "careers curriculum" to undergraduate students. At this writing the college-level version of the DISCOVER system has undergone only preliminary field trials. Under the sponsorship of the Exxon Education Foundation a carefully controlled field test of the system is scheduled to begin in September 1977. It is expected that this field test will provide answers to the following questions.

1. Do liberal arts students respond favorably to the computerized delivery of such a "careers curriculum?"

2. Is the operation of such a service cost feasible?

3. Will college and university faculty members accept and endorse a "careers curriculum" and more particularly, the computerized delivery of such a curriculum?

4. What effects will the system have on users? It is expected that students who use the DISCOVER system will report some or all of the following effects:

(1) They will show an increase in knowledge and awareness of personal values, competencies, and interests.

(2) They will show a significant increase in cognitive knowledge about and awareness of Holland's six occupational clusters.

(3) They will show an increase in career decision-making and planning skill.

(4) They will show significant movement toward specification of both educational and career goals.

(5) They will show an increase in vocational maturity.

(6) They will show an increase in vocational exploratory behavior.

Colleges and universities in increasing numbers are looking to computer technology as a means of holding the line on rapidly increasing personnel services costs. Many institutions are reluctant to become involved in the innovative use of the computer for the delivery of a "careers curriculum" because of the lack of quality information about costs, effectiveness, and student acceptance. The 1977 field test of DISCOVER will be a significant step toward the development of a "Computerized Careers Curriculum."

## References

Bowlsbey, J. H., Rayman, J. R. & Bryson, D. L. *DISCOVER Field Trial Report*. Westminster, Maryland: DISCOVER Foundation, Inc.; Western Maryland College, 1976.

Carney, Clark G. and Barak, Azy. A Survey of Student Needs and Student Personnel Services. *Journal of College Student Personnel*, 1976, (July), *17*(4), 280-284.

Chapman, W., Norris, L. & Katz, M. *SIGI: Report of a Pilot Study Under Field Conditions*. Princeton, New Jersey: Educational Testing Service, 1973.

Department of Defense. *Military-Civilian Occupational Source Book*. First Edition, July 1975. DOD1304.124.

Harris, J. E. *Analysis of the Effects of a Computer-based Vocational Information System on Selected Aspects of Vocational Planning*. Unpublished doctoral dissertation. Northern Illinois University, DeKalb, 1972.

Holland, J. L. *The Self-Directed Search*. Palo Alto, California: Consulting Psychologists Press, 1972.

Holland, J. L. *Making Vocational Choices: A Theory of Careers*. Englewood Cliffs, New Jersey: Prentice-Hall, 1973.

Hoyt, K. B. *Career Education: Contributions to an Evolving Concept*. Salt Lake City, Utah: Olympus Publishing Company, 1975, pp. 374-384.

Impelleteri, J. T. *Computer-assisted Occupational Guidance: The Development and Evaluation of a Pilot Computer-assisted Occupational Guidance Program*. (Final report) University Park: Pennsylvania State University, Vocational Education Department, 1968.

Maola, J. F. *An Assessment of Career Information Between OWE Students Using a Counselor-based vs. Computer-based Information System*. Unpublished doctoral dissertation. The University of Akron, Akron, Ohio, 1974.

McGrath, L. *Careers, Values and General Education. Liberal Education*, 1974, 60(3), 281-303.

Melhus, G. E. *Computer-assisted Vocational Choice Compared with Traditional Vocational Counseling*. Unpublished doctoral dissertation, Illinois Institute of Technology, Chicago, Illinois, 1971.

Myers, R. A., Lindeman, R. H., Forrest, D. J. & Super, D. E. *Preliminary Report: Assessment of the First Year of Use of the Education and Career Exploration System in Secondary Schools of Genesee County, Michigan*. New York: Teachers College, Columbia University, 1971.

Price, G. E. *A Comparison of Computer and Counselor Effectiveness in Assisting High School Students Explore and Select Courses*. Unpublished doctoral dissertation, Michigan State University, East Lansing, Michigan, 1971.

Pyle, K. R. & Stripling, R. O. *The Counselor, the Computer, and Career Development. The Vocational Guidance Quarterly*, 1976, *25*, 71-75.

Rayman, J. R. & Super, D. E. *Survey of Career Development*. Instrument developed for use as a part of the DISCOVER system. Westminster Maryland: DISCOVER Foundation, Inc., 1975.

Super, D. E., Stareshevsky, R., Mattin, N., and Jordann, T. P. *Career Development: Self-concept Theory*. Research Monograph No. 4. New York: College Entrance Examination Board, 1963.

Super, D. E. *Career Decision Tree*. New York: Teachers College, Columbia University, 1971.

Tiedeman, D. V. and O'Hara, R. P. *Career Development: Choice and Adjustment*. New York, New York: College Entrance Examination Board, 1963.

# ICS: An Intelligent Vocational Guidance Counseling System

**Paul Lorton, Jr.**
*Center for Business Education*
*and Research*
*University of San Francisco*

**Michael C. Hinckley**[1]
*IMSSS*
*Ventura Hall*
*Stanford University*
*Stanford, California 94305*
*(415) 497-2954*

**Robert L. Smith**
*Department of Computer Science*
*Rutgers University*

## Introduction and Focus

Computerized Vocational Guidance systems hold the promise of providing a breath and depth of information for which counselors are not suited. The main reason for this is that significant portions of vocational guidance are ideally suited for a very fast, accurate clerk. Tremendous amounts of information must be kept close at hand and transmitted to the inquiring student when needed. Further, through the developments in computer-assisted instruction and artificial intelligence, the computer can build a model of the student accurately and retain it indefinitely, permitting each session to build on the last.

These potential benefits have not yet been fully realized for several reasons. But at the Institute for Mathematical Studies in the Social Sciences (IMSSS), Stanford University, we are developing an individualized counseling system (ICS) to meet the promise of computerized guidance systems.[2] The system integrates advanced techniques from computer-assisted instruction and computer science into an interactive computer system for vocational guidance counseling. The design of ICS incorporates innovations and improvements in computer models of student interest and ability, program control, and structure of dynamic data bases.

## Objective of ICS

We have surveyed many implemented computer systems for vocational guidance and counseling. The main deficiency of these systems is that they do not use the symbol processing capabilities of the computer to optimize the individualization of the guidance process. A second problem is that some of the best systems fail to be transportable due to hardware and software constraints, and thus limit their national impact.

The ICS system seeks to individualize and modularize the guidance process by providing: (1) better computer models of individual students, (2) conceptual treatment of dynamic data bases for guidance, and (3) increased transportability of implemented systems. Each of these points is discussed below.

### *Computer Model of the Student*

The guidance systems we have surveyed all stress the importance of individualizing the guidance process and offer the prospect that computer-assisted guidance could do this. For example, Katz et. al. (1972) speak strongly for individual freedom and control in the SIGI system. Tiedeman (1970) deals with this problem in the ISVD system. A difficulty in some current programs is the roteness of the student's interaction with the computer. "The appearance of determinism" is addressed by Super (1970) where "the flow 'if a then b, if b then c' makes the outcome seem inevitable." An objective of ICS is to go beyond these initial efforts by using optimization and individualization techniques from computer-assisted instruction.

### *Dynamic Data Bases*

The issue of current and accurate data bases is obviously of prime concern for computer-based systems for vocational guidance. As Kroll notes "their effectiveness is dependent upon dynamic, ever-changing data files. It [the computer] is powerless to create data that heretofore have not been assembled or compiled in different forms." Of course the computer cannot create data from thin air. But data representation need not be limited to variations of lookup tables. More useful and sophisticated representations of data, including more powerful methods of making inferences from data, do exist and are being investigated in the ICS project.

### *Transportability*

"Computer-based guidance systems will probably never become a reality in education until the same program can be used in many locations," according to Kroll (1973). Within the hardware and software constraints discussed in later sections, the ICS project seeks to maintain transportability.

## Expected Benefits

The main benefit to be realized from the ICS project is a demonstration of how sophisticated computer models of data representation can be used to individualize the vocational guidance process in the context of computer-assisted guidance. The benefit for the individual student will be an increase in flexibility and pertinence of the dialogue with the computer.

Harris (1972, p. 9) points out that "the present types of computer-based guidance systems are in a very early stage of development," and that computer "technology is already well ahead of innovative guidance thinking." We agree with this statement. However, she goes on to say that "the refinement and expansion of the involvement of computers in guidance will depend more upon the innovativeness and dedication of practicing counselors than upon systems specialists. We find ourselves in some disagreement with this remark.

We feel that what is needed is the dedication and innovativeness of computer specialists who can comprehend the needs of the practicing guidance counselor and translate those needs into an improved and articulated guidance application. At IMSSS, we are capitalizing on a great deal of experience working with dedicated professionals in educational specialties, such as handicapped and minority instruction. It is our experience that we have something to offer these specialists, but that to be effective there must be a genuine dialogue. So we have regular contact with practicing guidance counselors which helps us to maintain a firm sense of what is actually needed and what will be most useful.

ICS seeks to improve the degree of individuality that the computer-assisted guidance system offers the student by improving both the model that the program has of the student and the model of the substantive guidance data base. In this effort, we are using computer techniques that have proved successful in individualizing computer-assisted instruction (CAI).

A related goal of ICS is to provide student-directed dialogue, and through this dialogue to increase student control over the counseling process. In discussing directions for future guidance systems, Harris (1972, p. 10) mentions the need for programs "which allow the counselee to prepare his own instructions to the computer and thus alter the predesigned processes." These two things are closely related, since the ability of the program to "alter the predesigned processes" in a useful way depends on the representation of the student's interests.

Consider the following scenario from the CVIS system (Harris, 1972):

Computer: Hi, Bill Brown. That is your name, isn't it?

Student: Yes.

Computer: The last time you were here, you were exploring occupations in the category of science that require community college training. What would you like to do today? Your wish is my command.

1. Continue exploration of this list of occupations.
2. Make new decisions about training.
3. Make new decisions about field of interest.

Student: Three.

Computer: Please choose one of the following. (Displays list of eight categories of interest.)

Student: Technology.

Computer: You showed significant interest in this field on the last interest inventory you took.

This dialogue reflects the linear nature of the program. A simple branching mechanism controls the student-computer exchange, and the student has no opportunity to pop-out to a higher level, establish his own flow, or ask questions according to his interest and needs. Likewise, the model of the student for this dialogue could be quite simple—a few variables to count interest in various occupational and educational features of a data base which is in reality more complex and interrelated than the program allows.

The restrictive design of the encounter is reminiscent of programmed instruction texts in the past. In fact, some guidance systems need not be implemented on the computer at all. Since they can be written out as text scripts in a manner similar to programmed instruction. We are not questioning the subject-matter expertise in the CVIS system. What we are arguing for is the application of the same technology to vocational guidance systems which transformed the notion of programmed instruction into the sophisticated and flexible computer-assisted instruction systems now in existence.

## The Components of ICS

The major components of ICS are a network model of the student, a network model of the guidance data base, and a MATCH component which relates the two models in a useful way. These components are shown in Figure 1, page 84.

### *Network Model of the Student*

Instead of modeling the student's interests and occupational goals in a relatively simple data structure, we are constructing networks that represent those interests. These networks contain nodes relating to occupations, fields, kinds of work—all interconnected in a manner that shows the conceptual relation these things have to the student. This Student Interest Network (SINE) may be seen in Figure 1. A general discussion of the use of networks in the computer representation of human knowledge and belief can be found in Simmons (1973), Schank (1973), and Abelson (1973).

### *Network Representation of the Guidance Data Base*

The ICS system also contains a network with a simplified model of the complete guidance data base incorporated in the program. The original guidance data base, as we have stressed, is obtained from currently existing systems. The Guidance Data Network (GDN) can be thought of as a guide to the data base, simpler than the complete data base, but more complex than an index or table of contents would be. The purpose of the GDN is to model the complete guidance data base for comparison to the network that represents the student. This comparison is performed by the network matching component MATCH. For a discussion of the uses of pattern matching in computer representation of knowledge see Wilks (1973).

Networks have been used to optimize CAI instruction, and this successful use is part of the reason we are applying networks to ICS. For example, the Basic Instructional Program (BIP) at IMSSS (Barr, Beard, and Atkinson 1975) uses a model of the student's acquisition of programming skills and concepts to select exercises for the student. In BIP, the exercises of the curriculum are also connected in a network. By comparing the model of the student with features of the stored curriculum network the exercises are selected.

### *Comparison of Student and Guidance Networks*

The MATCH component compares the SINE (Student Interest Network) and the GDN (Guidance Data Network) and provides the basis for selecting the questions and responses that ICS presents to the student. The uses of this comparison include:

1. Optimizing the questions asked by ICS, so that the greatest amount of information is obtained in the smallest number of questions. This corresponds strongly to the use of optimization techniques in the reading project at Stanford (Atkinson, Fletcher, Lindsay, Campbell, and Barr 1973), where items from the reading curriculum are selected to reduce the number of items the student must see in order to attain a given ability in the various categories ("strands") of the curriculum.

2. Presenting the student with information most closely related to the areas of his interest and skills.

3. Providing the student with information, within general areas of interest, which strengthens his conceptions about the occupations involved. For example, suppose a student shows interest in medical careers, as evidenced by his saying that he wants to become a doctor, but his network indicates an incomplete understanding of the structure of the medical careers available. It may then be appropriate to provide more information, not only by presenting specific facts, but also by giving the student a firmer sense of the structure of the medical fields, educational requirements, and so on.

Most current computer-assisted guidance systems use a simple table lookup and search strategy. We stress that alternatives to tables as representations of knowledge and to simple search strategies must be employed for truly individualized guidance. Recent work in the field of artificial intelligence has suggested that computer programs can interact carefully with humans only if they have detailed semantic models of the interaction process (Winograd, 1971).

### *The Dynamic Data Base*

One of the problems typically mentioned about computer-assisted guidance systems is the difficulty in changing and updating the data base. This is seen as a yearly task of installing new tables and updating the program where necessary (Kroll 1973; Harris 1972). The past several years have seen enormous structural changes in economic patterns and job opportunities, as well as the value systems of students. (See Katz [1973] for a discussion of the role that values play in counseling in the SIGI system.) Such changes may not be representable in the mere updating of parameters and tables. ICS is designed with flexible data structures to permit ready, and nearly automatic, updating by means of interactive data management routines, which are an integral part of the program.

## Transportability

We recognize the desirability that the research and implementation involved in ICS be as transferable as possible to other educational settings and systems. One associated problem already mentioned is that the data base may contain geographically local information and requires periodic updating.

The issue of the transferability of guidance programs is considered in detail by Kroll (1973), who points out that the large diversity of hardware and software often precludes implementation on a given system of a program written elsewhere. This problem is not specific to vocational guidance systems, or even more generally to the uses of computers in education, and hence is not really "solvable" by a project of the scope of ICS. There are too many broader and more complex issues at stake, including hardware and software compatibility, telecommunications and networks, and the issue of large centrally located computers versus small, on-site minicomputers. While we cannot provide a definitive solution to the problem, we recognize its importance. Our objective is to anticipate the form of likely solutions and we are trying to provide the most reasonable implementation under the current uncertainties.

### *Software Compatibility*

Guidance systems written in machine language clearly have the most limited transferability because they depend on the specific machines for which they were designed. Currently, OIAS (Kroll, 1973) is written in IBM 360 machine language, and SIGI (Godwin, 1974) is written in machine language for a dedicated PDP-11.

Languages such as BASIC and FORTRAN are widely implemented on a variety of large and small machines. Currently, the Guidance Information System, marketed by Time Share Corporation, is written in both FORTRAN IV and BASIC. It runs on some medium size (PDP-11) and large machines (PDP-10), provided the proper disk system is available. However, these languages are primarily intended for numerical rather than symbolic processing, and have poor control structures since they generally lack recursion.

We have chosen to implement ICS in LISP (McCarthy, 1969). We are using a neutral dialect of LISP that will be transportable with little difficulty to many larger machines, such as IBM 360-370 and CDC 6600, as well as some smaller machines. More importantly, however, the choice of LISP is the best language for implementing ICS, given the constraints currently facing such projects.

1. The computational complexity of the ICS system currently requires a relatively large machine (the size of a PDP-11 or larger) regardless of which high-level language is chosen. So an implementation in FORTRAN or BASIC would not dramatically increase the immediate transferability of ICS to smaller machines, unless it was machine coded.

2. LISP is the language most frequently chosen when data structures become complex and difficult to predict in advance. As a language it is relatively stable and widely used.

3. The prospects for later transfer of ICS and the ideas that will be incorporated into it are excellent. This is because, whatever solutions are found to the general problems of hardware-software compatibility, algorithms designed in LISP will have a great influence on the areas for which they are designed. If ICS succeeds in its goals, the work will be transportable.

### *Hardware*

No special hardware is required to run ICS. The mainframe machine must be capable of supporting LISP or an equivalent system, and a high-speed disk for the data must be available interactively.

The peripheral constraints are minimal. ICS is designed to run on any full-duplex ASCII terminal, hard copy or CRT, with appropriate code for both contingencies. We will be using CRT displays running at 1200 baud for high-speed delivery. These terminals are standard devices that currently market for about $2,200 each. We have chosen this method of information delivery for a comprehensive system over other media (e.g., slides, films, etc.) because of the basic reliability and serviceability of these display units as demonstrated at the IMSSS computer laboratory and in IMSSS field projects.

## Cost Effectiveness

The Institute has always stressed the need for cost-effectiveness when using computers in education (Jamison et. al., 1971), and IMSSS has instituted many internal measures to optimize the use of its facilities. As a result, even in the trial phase of the project, the system will be nearly as cost effective as systems already in production use. For example, the terminal cost for a month during the second period will be less than $1,000, including the cost of a phone line.[3] Estimating the effective use of a terminal to be six hours per school day, this amounts to approximately nine dollars per student contact hour. As a comparison to a commercially available system, the Time Share Corporation GIS system is said by Kroll (1973) to cost six to twenty dollars per hour, using Model 33 Teletype terminals at 110 baud for the delivery. We estimate that we could provide delivery on a production basis for less than the lowest amount quoted for the TYMSHARE system, i.e., six dollars per contract hour.

## Conclusions

Our growing experience with ICS is producing a system for guidance which is meeting the promise of computerized guidance systems. We are developing a data base management system which is handling the task. Nearly automatic updating of information is a reality in many important respects. Finally, the student modeling effort is producing a more responsive counseling engagement with the student.

At a time when the orientation of college students is toward the world of work and the world of work is changing rapidly, the realization of the promise of computerized guidance systems is even more necessary. This report is of a project which fits this need and which is proving successful in meeting the promise of computerized guidance systems.

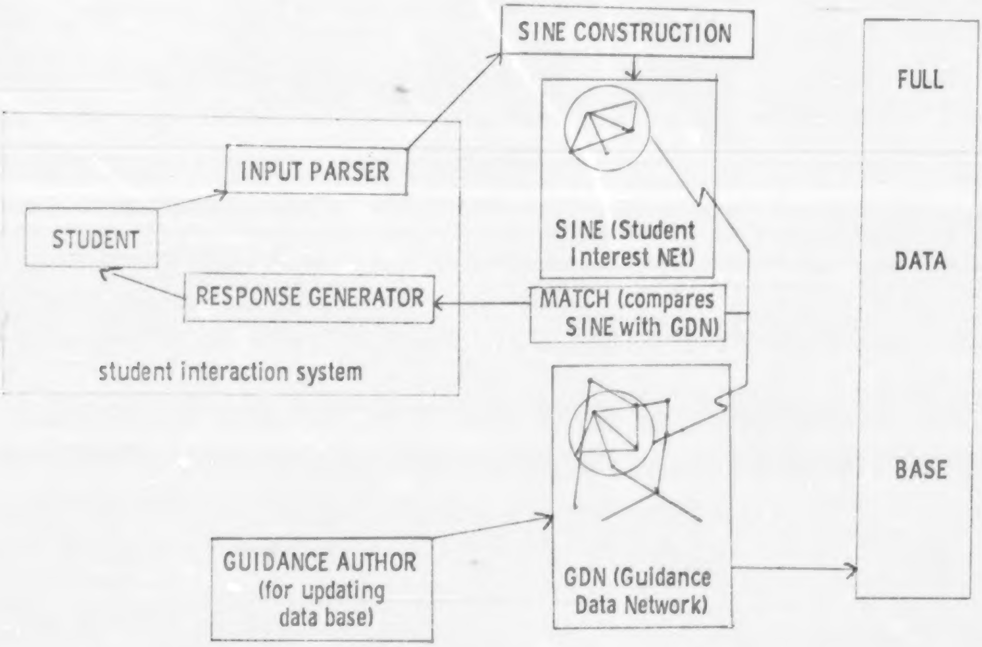

SINE CONSTRUCTION
FULL
DATA
BASE
INPUT PARSER
STUDENT
SINE (Student Interest NEt)
RESPONSE GENERATOR
MATCH (compares SINE with GDN)
student interaction system
GUIDANCE AUTHOR (for updating data base)
GDN (Guidance Data Network)

Figure 1. CIS System Components

## Notes

1. Correspondence with regard to the contents of this paper should be directed to Professor Hinckley.

2. The authors would like to thank Professor Patrick Suppes for his able direction of this project. The work described here is being supported under Department of Health, Education, and Welfare Office of Education grant G007603721.

3. By using Model 33 teletpyes, we estimate that we could reduce the cost to about fifty dollars.

## References

Abelson, R. P. The structure of belief systems. In R. C. Schank and K. M. Colby (Eds.), Computer models of thought and language. San Francisco: Freeman, 1973.

R. C. Atkinson, J. D. Fletcher, E. J. Lindsay, J. O. Campbell, and A. Barr. Computer-assisted instruction in initial reading. 1973.

Barr, A., Beard, M., and Atkinson, R. C. The computer as a tutorial laboratory: The Stanford BIP project (Tech. Rep. No. 260). Stanford, Calif.: Institute for Mathematical studies in the Social Sciences, Stanford University, 1975.

Godwin, W. SIGI: An operational example of computer-based career guidance. Princeton, N.J.: Educational Testing Service. RM-74-9, 1974.

Harris, J. Computer assisted guidance systems. National Vocational Guidance Association, 1972.

Jamison, D., Fletcher, J. D., Suppes, P., and Atkinson, R. C. Cost and performance of computer-assisted instruction for education of disadvantaged children. (Tech. Rep. No. 175). Stanford, Calif.: Institute for Mathematical Studies in the Social sciences, Stanford University, 1971.

Katz, M., Chapman, W., and Godwin, W. SIGI: A computer-based aid to career decision-making. Educom Bulletin, Summer Issue, 1972.

Kroll, A. Computer-based systems for career guidance and information: A status report. Princeton, N.J.: Educational Testing Service, 1973.

McCarthy, J. LISP 1.5 programmer's manual. Cambridge, Mass.: M.I.T. Press, 1969.

Shank, R. C. Identification of conceptualization underlying natural language. In R. C. Shank and K. M. Colby (Eds.), Computer models of thought and language. San Francisco: Freeman, 1973.

Simmons, R. F. Semantic networks: Their computation and use for understanding English sentences. In R. C. Shank and K. M. Colby (Eds.), Computer models of thought and language. San Francisco: Freeman, 1973.

Winograd, T. Procedures as a representation for data in a computer program for understanding natural language. MAC—TR-84. Artificial Intelligence Laboratory: Massachusetts Institute of Technology, February 1971.

# Computers and Teacher-Education

**Michael A. Lorber**[1] **and Steven J. Skinner**
*De Garmo Hall*
*Illinois State University*
*Normal, Illinois*
*(309) 436-6355*

## Introduction

Computer assisted instruction (CAI) can, and should, be an important part of teacher education. What follows is a brief description of a non-traditional teacher-preparation program, the reasons that program is increasing its use of CAI, and descriptions of the kinds of CAI being planned. Other teacher educators who are considering CAI for their programs may find the information useful.

## The Professional Sequence

The Professional Sequence is a self-paced, competency-based, teacher-preparation program for prospective secondary school teachers. The program is being implemented by the Department of Curriculum and Instruction at Illinois State University and has an average enrollment of about 1,200 students per semester.

The sequence enables students to learn at their own rates by dispensing with traditional classes and relying instead on a series of self-instructional packages which students complete at rates commensurate with their own abilities. These packages serve a dual role of facilitating self-pacing and introducing students to a variety of instructional tools. In many cases the packages are, themselves, new instructional tools to students, and most packages incorporate either films, videotapes, tape-slide programs, or seminars. Students also use audio and video tape recorders in analyzing their own teaching. A computerized recordkeeping system is used to keep track of which packages each student has completed, and no student graduates from the program until all required competencies have been demonstrated.

Although the Professional Sequence has proven to be at least as effective in preparing secondary school teachers as its more traditional counterparts,[2] it is continually striving to improve itself. One revision planned for the fall of 1977 is the incorporation of CAI.

## Why CAI?

A number of reasons for incorporating CAI into teacher-preparation programs are as applicable to traditional programs as they are to non-traditional programs. Consider the following points:

1. Almost all teacher-educators agree that they could do a better job preparing teachers if they had more time. To the extent this is true, CAI can help improve teacher-education by making available more instructional time. Many studies have shown that CAI can save instructional time while maintaining or increasing students achievement levels. One study, for example, ". . . showed that adults learning computer programming through CAI took from five to ten hours to reach the same level of proficiency as students who spent from twenty-four to thirty hours in traditional instruction."[3] Another study ". . . compared computer simulation of high school physics experiments with traditional laboratory experiments and found that the CAI students learned more in one-eighth the time."[4] Numerous other studies could be cited but they would verify the same point—students can spend significantly less time learning via CAI and still achieve as well as, or better than, students who learn via traditional methods.

2. Many teacher-preparation programs rely heavily on classroom activities such as lectures and discussions. While some topics

of interest to teacher-educators lend themselves readily to such instructional tools, other topics (such as the structure of instructional objectives, the classification of objectives according to taxonomic levels, and certain aspects of measurement and evaluation), may be taught more effectively via tutorial and drill and practice CAI programs. At the very least, such programs could reduce the amount of time spent on basic instruction about these topics and thus free more time for in-depth analyses of these and other topics.

CAI may also be of use in helping students develop their decision-making powers. In a classroom situation it is virtually impossible to enable students to make decisions and to provide each student with feedback concerning the consequences of those decisions. Simulation programs *can* do this and they can do it for numerous students simultaneously.

It seems apparent that if we teacher-educators want our graduates to go into their own classrooms and use a variety of instructional tools appropriately, we should set the example in teacher-education programs. We should, for example, teach more *with* CAI as well as more *about* CAI.

3. The American Institutes for Research (AIR) funded a study to determine how much growth there was in computer utilization by the nation's secondary schools from 1970 to 1975. The study showed that while about 34 percent of the schools used computers in 1970, ". . . over 58 percent of the nation's secondary schools today use a computer to aid their administrative or instructional programs."[5] The study also projected that if current trends continue, ". . . within the next decade every secondary school in the country will have access to a computer system for some type of administrative and/or instructional application."[6]

The AIR study pointed out that the most frequent instructional applications of computers were in courses related to computer science, business, and mathematics,[7] but given the increasing availability of computers and the continual search for ways to individualize instruction and help students develop decision-making powers, it seems realistic to predict an expansion in the use of CAI.

The fact that increasing numbers of schools are acquiring computers should be, itself, sufficient warning to teacher-educators to begin producing more graduates who can capitalize on the computer's potential. The warning becomes even more clear when one considers that schools are continuing to acquire computer power even now, during times of extreme financial hardship.[8] As administrators struggle to make fewer dollars do more work they will undoubtedly try to employ the most highly qualified and versatile teachers possible. If other factors are about equal, teacher candidates who have had experience with computers as instructional tools and who can help administrators make the best possible use of available computer power will likely have a decided advantage over candidates without this background and skill. The inclusion of CAI in teacher-preparation programs may thus be a deciding factor in helping graduates acquire teaching positions.

4. Introducing students to CAI is a little like giving them just one peanut—it makes them want more. In the Professional Sequence, for example, tutorial and drill and practice programs were field-tested with over 500 students, and over 98 percent of those students indicated they liked CAI and wanted more of it.

The interest students show in CAI carries over into other computer applications, particularly problem-solving and inquiry. Sequence students use the computer to generate Flanders' Interaction Analyses from data they acquired from observing classes, and for ERIC searches as well as for CAI. This kind of familiarity with an instructional tool is exactly what is needed by teachers if they are to use that tool to its best advantage in meeting the instructional needs of their own students.

In addition to finding out how computers can aid instruction, students should also find out how computers aid in scheduling, grade reporting, guidance, long-range planning, inventory control, and bookkeeping. This kind of information can help broaden students' perspectives and give them a more realistic idea of how modern school systems operate. This, in turn, can help students acquire a better understanding of education as a whole and their role within that whole.

5. "One of the goals shared by teacher educators is the production of a cadre of professionals who view methodological research as a scholarly activity worthy of their time and effort, and who are responsive to research results which suggest a reevaluation of classroom practices."[9] Other areas such as the sciences, psychology, and industrial technology are seizing the initiative. Their work with computers is helping them attract students because these areas are seen as fitting into the future. Few students want to participate in programs which appear provincial and even backward when they could participate in programs which seem to clearly prepare them for the demands of tomorrow. Teacher education programs should be among the most innovative and exciting programs on any college campus.

More to the point, if teacher-educators choose to ignore or minimize the potential of computers there is little reason to suppose prospective teachers will use computer power when they get into their own classrooms. This would be unfortunate not only because a tool of demonstrated effectiveness would lay unused, but also because some students who might profit from CAI will be denied that opportunity.

There are, of course, other reasons to include CAI in teacher-preparation programs (such as its cost-effectiveness[10] and its utility in in-service programs[11]), but these reasons simply reinforce the reasons already cited. Logic, as well as the facts cited, support decisions to include CAI as an integral part of teacher-preparation programs. As it happens, most teacher-preparation programs deal with some topics which lend themselves readily to CAI.

## CAI Topics for Teacher-Education

At the present time CAI in the teacher-preparation program at ISU focuses on five topics—(1) the structure of instructional objectives, (2) taxonomic divisions of objectives, (3) measurement and evaluation, (4) Flanders' interaction analysis, and (5) curriculum development. These topics provide vehicles for the use of drill and practice, tutorial, inquiry, simulation, and problem-solving programs and thereby give students some idea of CAI's flexibility and some basis for assessing CAI's usefulness as an instructional tool.

### The Structure of Precise Instructional Objectives

The objective of this program is to enable students to specify, via the computer, what structural weaknesses there are in at least 80 percent of a given series of objectives.

Students begin by inputting their names and then opting either to review the structure of precise instructional objectives or to bypass the review. The review is extensive enough to enable students to be successful in the remainder of the program even if they have little previous knowledge of objectives. The review is presented via a tutorial format.

Once students either go through, or bypass, the review they are presented (via a modified drill and practice format) with a series of objectives and a constant set of weaknesses common to objectives (such as lacks an observable behavior or lacks a minimum acceptable standard). Corrective feedback and/or reinforcement is provided following each student response, and following completion of the first series of objectives the program prints out a list of which objectives were missed and analyzes the student's response pattern.

If students miss more than 20 percent of the items they receive remediation based on the analysis of their areas of weakness. Following the remediation, students are presented with a second series of objectives and the cycle of response, feedback, and analysis is repeated. If students miss more than 20 percent of the items this time they are referred to a faculty member for further help. Students achieving the 80 percent criterion level of the first set of objectives have the option of doing or bypassing the second set.

The program is written in BASIC and takes students an average of thirty minutes to complete.

### The Cognitive Domain

The objective of this program is to enable students to specify, via the computer, the cognitive level of at least 80 percent of a given series of objectives. Although the content of this program deals with differentiating levels within the cognitive domain, the programming parallels the previous program. There is an extensive, but optional, review utilizing a tutorial format followed by two series of objectives with a response-feedback, analysis cycle utilizing a modified drill, and practice format. This program is also written in BASIC and takes students an average of thirty minutes to complete.

### Measurement and Evaluation

The objective of this program is to enable students to select the definitions and/or apply, via the computer, at least 80 percent of a given series of terms common to measurement and evaluation.

Students begin this program by inputting their names. The program then prints out a list of twenty terms basic to measurement and evaluation, and students, using an inquiry format, can get definitions and explanations of any or all presented with a series of situations in which they must apply or define the terms. Corrective feedback and/or reinforcement is provided following each response.

After students complete the modified tutorial section, they receive an accounting of the items they missed and an analysis of their weaknesses. They are told, for example, if they are weak in the general area of reliability, validity, and overall measurement and evaluation; the general area of kinds of tests; and/or the general area of statistics. Students are also given the option of an additional review of the area(s) in which they are weak.

The program is written in BASIC and averages about thirty minutes in length.

### Flanders' Interaction Analysis

At the request of a Professional Sequence faculty member (Dr. Robert Fisher), a problem-solving program was constructed which enables students to input the data they record as they chart verbal interaction in a class and receive a full analysis of the data via computer printout. Since the program calculates interaction matrices and makes various correlations, students are more willing to use Flanders' interaction analysis. Further, availability of the program has enabled the sequence faculty to build Flanders' interaction analysis into a number of micro-teaching packages.

The program is written in FORTRAN and takes students about ten minutes to use.

### Curriculum Development

To date, the most ambitious CAI application in the Professional Sequence is in the area of curriculum development. At the request, and with the assistance, of Dr. Clifford Edwards three CAI programs were developed to help students understand the Edwards' decision-making model and to give them practice in using that model.

In the first program students opt either to receive or to bypass an extensive review of the Edwards' decision-making model. Continuing in a tutorial format students then "walk through" a hypothetical situation using the model and receive feedback concerning the interaction of their various en-route decisions.

The second and third programs in the series are two parts of a single simulation program in which students apply the Edwards' decision-making model to a given situation and receive feedback concerning the internal logic and consequences of their decisions.

The programs are written in BASIC and take students about forty minutes each to complete. Revisions are planned to reduce completion time to about thirty minutes each.

## Conclusion

Given the increasing numbers of schools acquiring computer power and the demonstrated effectiveness of CAI, it would be appropriate for teacher educators to make CAI an integral part of their programs. In fact, it is reasonable to assume that those teacher-educators who are slow to incorporate CAI into their programs will begin experiencing declines in their enrollments as prospective teachers seek out more progressive and up-to-date programs. this would be unfortunate because the move toward CAI is neither difficult nor painful. CAI belongs to today's teacher-preparation programs. Help put it there.

## NOTES

1. All correspondence concerning this paper should be directed to Professor Lorber.

2. Dennis Gene Wiseman, "A Criterion-Based Comparison of Student Teachers From Two Teacher Education Programs, One Competency-Based and One Conventional," Unpublished doctoral dissertation, University of Illinois at Urbana-Champaign, 1974.

3. Marcella L. Kysilka, "How Effective is CAI? A Review of the Research," *Educational Leadership*, November 1975, pp. 149-150.

4. *Ibid.* p. 150.

5. William J. Bukowski and Arthur L. Korotkin, "Computing Activities in Secondary Education," *Educational Technology*, January 1976, p.9.

6. *Ibid.* p. 20.

7. *Ibid.* p. 9.

8. "Education—Learning to Live With Less," *Time*, October 4, 1976, pp. 58-63.

9. Gary D. Brooks, "Computer Science: A Neglected Area in Schools of Education," *Phi Delta Kappan*, October 1971, p. 121.

10. Judith Edwards, et. al., "Is CAI Effective?" *AEDS Journal*, Summer, 1974, p. 124.

11. Keith A. Hall et. al., "A Triumph for CAI," *Phi Delta Kappan*, September 1974, pp. 70-72.

## References

Brooks, Gary D., "Computer Science: A Neglected Area in Schools of Education," *Phi Delta Kappan*, Vol. LIII, No. 2, October 1971.

Bukoski, William J., and Arthur L. Korotkin, "Computing Activities in Secondary Education," *Educational Technology*, Vol. XVI, No. 1, January 1976.

Churchill, Geoffrey, "Computers in the Undergraduate Curriculum: An Aspect of the Many Section Problem," *AEDS* Journal (Association for Educational Data Systems), Vol. 9, No. 3, Spring, 1976.

Edwards, Judith, et. al., "Is CAI Effective?" *AEDS Journal*, Vol. 7, No. 4, Summer, 1974.

Hall, Keith A., G. Phillip Cartwright, and Harold E. Mitzell, "A Triumph for CAI," *Phi Delta Kappan*, Vol. LVI, No. 1, September 1974.

Hedges, William D., "The Principal, the Computer, and Emerging Applications to Instruction," *Phi Delta Kappan*, Vol. LV, No. 3, November 1973.

Rockart, John Fralick, and Michael S. Scott Morton, *Computers and the Learning Process in Higher Education*—A Report Prepared for The Carnegie Commission on Higher Education, New York: McGraw Hill Book Company, 1975.

Spencer, Donald D., "The Computer Goes to School," *AEDS Journal*, Vol. 6, No. 1, Fall, 1972.

Wiseman, Dennis Gene, "A Criterion-Based Comparison of Student Teachers From Two Teacher Education Programs, One Competency-Based and One Conventional," Unpublished doctoral dissertation, University of Illinois at Urbana-Champaign, 1974.

"Education—Learning to Live With Less," *Time*, October 4, 1976.

# Report on a Highly Used Computer Aid for the Professor in his Grade and Record Keeping Tasks

**Richard Brockmeier**
*Hope College*
*Holland, Michigan 49423*
*(616) 392-5111 Ext. 2308*

## Introduction

A data base management system called SPARS for entering and processing a professor's student data has been developed in the physics department at Hope College. The package is designed to operate in both batch and interactive modes on Xerox Sigma 6 computer requiring 14K words of memory. It is the purpose of this paper to describe the design philosophy and some of the features of this package. The results of our experience with its use at Hope College and future plans are to be discussed.

## Design of SPARS

SPARS is a data base manager and processor package designed to aid the professor in handling his students' grades and other classroom data. It can handle multiple sections, labs, etc., and allow the professor to combine and separate these components in a variety of ways. It seeks to meet the sometimes competing goals of simplicity of use and high security.

The limits on the number of independent classes per professor, number of students per class, and data per student are dynamic and can be changed as the professor's needs change; the absolute limits are generous enough not to be of concern to the professor (i.e., up to 1,000 classes per professor, up to 5,000 students per class, and up to 256 words per student). The professor references the class by a class code, the student by student number, and data elements within each student's record by means of one to three character variable-names which the professor designates at the time he creates the variable. Information about these data elements is stored in a Variable Dictionary automatically by SPARS and includes a description of the variable, what width of field should be used whenever it is printed out, and what type of conversion to use. The professor retains control of the spacing and placement of items in his printouts, but is relieved of chores of formatting. The data itself is contained in student records. These facts however are transparent to the professor. He does not need to know the name of the file, its password, or its structure. He merely LOGS-ON to SPARS and identifies his class-code together with some security identification procedures.

The program is structured with an executive which recognizes commands and then branches into the appropriate routine to handle the request. For clarity, the commands can be grouped (according to type of function) into the several different classes which are shown in Table 1. Tables 2 through 9 then show the individual commands within each of the classes together with a brief description of the purpose of the command. In addition to the information presented in the tables, it should be pointed out that all commands begin with either an "*" or a "?". All commands beginning with the "*" perform some operation, whereas all commands beginning with "?" perform no operations but merely give extended on-line documentation to the user. For example, for every *-command, there is a companion ?-command explaining the *-command. In addition, many error messages will refer the user to a ?-command for a more lengthy description of what the problem is. SPARS consists of 41 *-commands and about 1,000 lines of documentation accessible through ?-commands.

We shall now briefly itemize a representative sample of some of the features which have been designed into SPARS.

1. The user does not need to explicitly create a variable entry in the Variable Dictionary prior to putting data into the variable. SPARS will create the new entry automatically based on whatever information it already has on the variable. The user can later modify any of the parameters with the *DEFINE command if he is unsatisfied with the automatic result.

2. Unlike the storage of conventional FORTRAN variables, prior to the storage of any data in a SPARS variable, the location is set = to a "value" of ?. (e.g., if you requested SPARS to printout the values of test scores, you would get question marks printed out for any students you skipped over.) Further, if the user so wishes, he can enter a "blank" into a variable location for a student (to indicate that a student didn't take a test, etc.). Any subsequent processing such as averages, correlations, etc., ignores "blanks" and "question-marks." Hence, the problem of dealing with missing data is handled automatically for the user.

3. Unlike conventional FORTRAN, one of the DATA-TYPES is the G-type conversion which accepts letter-type grades (A,A-,B+, etc.) and converts them for internal storage as numeric values (40, 37, 33, etc.). Hence, all processing commands such as *AVERAGE, *CORRELATION, *WAVG (weighted average) proceed in a normal fashion. In fact, this allows an easy means for combining both numeric scores and letter-type grades for arriving at a final raw score. Upon output, G-conversion converts the numeric results back into a letter grade in the printout.

4. As a convenience for the professor (and also as a security feature), SPARS automatically enters into the Variable Dictionary the time, the date, and a code indicating via which command a variable was created. It also enters the time, date, and code for any changes which were made to the data stored under each variable.

5. The professor can request his class printouts to occur in any sorted order of any of the variables, nested up to three deep. Submitting the *ORDER command produces a chaining pointer which is stored in the student's record as a transparent new variable which the user cannot reference directly. If one submits the *WRITE command, SPARS will use the transparent chaining variable which was created or referred to by the most recent *ORDER command to produce the printout of the class in the desired order. Any number of sorts can be stored (restricted only by the 256 maximum limit to the number of variables). Hence, future requests for a previously sorted order will not require a fresh sort. However, any change of data which affects the order of any existing sort flags it as not valid, forcing a new sort at the next time it is requested and the transparent variable is deleted automatically the next time the user enters the *DELETE command. Thus, the professor can get multiple copies of his class—by seating chart, alphabetical by name, from highest to lowest on any test, by lab partner in the lab, etc.

Whenever the professor wants to operate on the data, he must communicate to SPARS a list of variable names to be processed or printed out. A convention under SPARS requires that parentheses initiate and terminate the list. The individual names are to be separated by commas or blanks. Two variable names separated by a hyphen indicate all existing variables which lie between the two in an alphabetical sense. Thus, if he wants to add the scores of six quizzes and ten lab experiments, and put the results in variable RAW, he would say: *ADD (Q1-Q6,E1-E10) (RAW).

## Security Provisions

The computer facility is operated in such a fashion at Hope College so as to provide its students with as much experience and direct contact with the computer as possible on a modest budget. Students can frequently be found in the operating room. In fact, all evening operations are conducted with student help with no college personnel at the center. Some thought must be given to security. The philosophy followed has been that: (1) the name, account, and password of the file containing the student data will become known, (2) a User Manual for SPARS will accidently be left lying around by a user, (3) there will be those who will be "curious," and (4) playing games with students should be avoided. At the same time, the system must be sufficiently simple so that the professor with no prior computer experience will not be discouraged from using it. Moreover, SPARS must contain its own security system, and the source should be highly protected.

Such features as encryption of bits in the record words and extensive indirect addressing of data elements are used to mix up the files, but add no complications to ease of use from the user's point of view. Therefore, one can not make sense of the files except through SPARS itself. Additionally, the system knows which users and user accounts are valid, can tell if a file has been copied and then accessed from a foreign (but still valid) account, etc. Means for recording such occurrences are included without letting the unauthorized person know he has been detected. The professor has the option of causing the sign-on protocol to change if added security is desired.

## Experience with SPARS

There has been a steady growth in the use of SPARS since its inception at Hope College. At present, approximately 8-10 percent of the student credit hours are processed by professors through SPARS at Hope College. It was developed and initially used solely in the physics department by faculty who were highly proficient in using the computer. More recently, its use has spread to professors who have had no prior experience with computers. Also, secretaries have quickly learned how to use a terminal and have entered data for professors. This has proven to be a positive initial experience with the computer—something which several confessed was a surprise. This did not come without a number of earlier unsatisfactory experiences, however.

An earlier version of SPARS lacked some of the present automatic features and didn't have on-line documentation. Several professors with no prior computer experience attempted to use this earlier version. Although they entered with enthusiasm, they soon became stuck on one minor point after another, and ultimately surrendered to a feeling of frustration and wasted time.

But the experience was a valuable one. By watching several non-computer users who were willing to allow you to observe their interactions with the package, this writer was able to make some major changes which very significantly improved the package.

The package is now used in the departments of physics, computer science, chemistry, biology, psychology, sociology, foreign languages, English, business and economics. Professors without prior computer experience tend to restrict their use to the rather basic commands which parallel procedures one would follow in manual grade-keeping. Many professors find that just doing this much on SPARS is of value for classes of twenty-five or more students.

Initially, they tend not to grasp the flexibility which computer data management under SPARS can bring to the process of evaluating one's students. With repeated use, many are beginning to discover its capabilities for doing new analyses like computing, z-scores, studying correlations, etc. In the process, this has brought changes to the way some courses are taught and managed. In the physics department, we use SPARS quite extensively to handle the beginning physics sections and labs. SPARS is designed to receive grades from multiple graders, normalize the scores and sort them out. In addition, it has been used to aid in selecting students for the honors section.

## Conclusions

It is our experience at Hope College that professors with no prior computer experience will use the computer when grade handling packages such as SPARS are made readily available and are sufficiently flexible and easy to learn and use. The use of SPARS at Hope College appears to be steadily growing without administrative push or encouragement. In seeking ways of developing the computer as an aid in the process of education, the major emphasis has been placed on interactions between the *student* and the computer in CAI, computer generated tests, self-paced study, etc. Our experience with SPARS at Hope College indicates that efforts to interface the *professor* with the computer in the more traditional classroom and laboratory environment has not yet been fully explored. It may also be true that by giving a professor an easy and flexible tool like SPARS for intermixing grades and scores from a variety of sources including the traditional ones, we may be smoothing the way for his adoption of CAI units and other forms of computer aids. This seems probable for two reasons: (1) Through SPARS he has developed a comfortable familiarity with the computer himself, and (2) with SPARS he has an easy and convenient way for intermixing grades, scores, and information from a variety of sources including the traditional ones.

The next step at Hope College, should the use of SPARS continue to grow, will be to develop additional commands and more automatic procedures for setting up class files with the names and student numbers at the beginning of the semester so that SPARS can be initiated by the professor via a simple command. Finally, some thought is being given to adding a feature which would enable the professor to enter a flag into the Variable Dictionary indicating his authorization to use the contents of that variable for student advising purposes. These variables would contain, for example, information on the student's most recent progress in class. Another package could be designed to search the SPARS files to identify students, particularly freshmen, who are beginning to have academic difficulty so that they could be placed in contact with their advisors prior to midterm. Hopefully, one could get these students into a more suitable class schedule while it is still possible, and save them from losing a semester or becoming dropouts.

Since SPARS is written in FORTRAN, the possibilities of it being transportable to other sites are good with additional work. Some features of SPARS depend upon the word size being thirty-two bits or greater. Other features depend upon options available under the CP-V operating system of XEROX computers which would have to be supplied in some fashion on a different machine. If outside interest develops, the possibilities of transfer to other machines could be explored.

**Table 1**

**CLASSES OF COMMANDS**

| | |
|---|---|
| ASSISTANCE COMMANDS | (Includes on-line documentation for the user) |
| CONTROL COMMANDS | (These are global commands which affect operation of other commands) |
| DATA INPUT COMMANDS | (Variety of means for entering data) |
| DATA OUTPUT COMMANDS | |
| HOUSEKEEPING COMMANDS | |
| PROCESS DATA AND PRINTOUT RESULTS COMMANDS | |
| PROCESS DATA AND STORE RESULTS COMMANDS | |
| SECURITY COMMANDS | |

## Table 2

## ASSISTANCE COMMANDS

| | |
|---|---|
| ?COMM[ANDS]: | Gives the names of all existing commands. |
| *GLOS[SARY]: | User can search the DICTIONARY of variables he has created and related parameters. |
| ?HELP: | Lists other ONLINE sources of help. |
| ?NEW: | Initiates a self-study program for learning SPARS. |
| ?SUGG[ESTION-BOX]: | User can direct questions/suggestions to the SPARS administrator. |
| ?xxxx: | Gives documentation about command *xxxx. See text. |

## Table 3

## CONTROL COMMANDS

*DATS[W]: For non-standard unadvertised features.

*ECHO: Used in BATCH; toggles flag controlling printout of input.

*INCL[UDE-IF]: Restricts all processing and printing to a subset of the class of students which user specifies until next *INCL command is received.

*LENG[TH],*WIDT[H]: Allow user to change page lengths or widths from default values.

*LPRI[NTER]: While in interactive mode, user can automatically direct all output (except prompts, error messages, and advisories) to the high-speed printer to reduce terminal time.

*ORDE[R]: User specifies student order of printout of all reports following this command until a new *ORDER is received. It is capable of sorting on any variable in file with nesting up to three levels.

*TOSS[OUT]: The user may control which data for students are to be ignored (not included) by the actions of all PROCESSING COMMANDS which follow this command.

*TITL[E]: Enters titles for reports.

*WARN: User arms/disarms the warning messages for (1) keying-in of non-existent student numbers and (2) whenever user enters data on-top-of existing data.

## Table 4

## DATA INPUT COMMANDS

| | |
|---|---|
| *CHAN[GE]: | For entering/correcting the datum stored in a variable for one student. |
| *GLAB: | For submitting data from student lab section grades. Used with the *LAGD command. |
| *PROM[PT]: | User enters data for students without need for keying in student numbers. User can request that he be prompted by student's name and that the student order be the same as any previously printed output. |
| *QUIZ: | Designed for entry of data from "quiz cards." Brief statistics are reported at the end. |
| *READ: | For entering multiple variable data keyed on the student number. |
| *SAME: | Similar to *QUIZ except no statistics are given. |

### Table 5

### DATA OUTPUT COMMANDS

| | |
|---|---|
| *LAGD: | Produces a deck of cards by student and by lab-experiment (or quiz, etc.—the professor specifies). Then the lab graders report the grades back to the professor via these cards. The professor submits these cards in BATCH after the *LAGD command. |
| *PUNC[H]: | Allows professor to send data to another file or periferial such as the card punch. |
| *WRIT[E]: | Prints class reports on variables specified by professor. The order of the students in printout will be controlled by the most recently submitted *ORDER command. The layout of the variables across the page is controlled by the contents of the VARIABLE DICTIONARY and in the order requested by the professor. Extra spaces between fields are generated if the professor places extra spaces between the variable names. Which students are included in the report is controlled by the most recently received *INCLUD-IF command. Multiple reports from just one command are possible. |

### Table 6

### HOUSEKEEPING COMMANDS

| | |
|---|---|
| *CHAN[GE]: | Enables professor to correct any entry for any student. This includes the status variable S which if given a value other than 1 will logically drop the student from the class file (student can be retrieved by resetting S to 1). |
| *DEFI[NE]: | Enables the professor to modify contents of the VARIABLE DICTIONARY such as variable definitions and field widths. He may also create new variables in the DICTIONARY for the first time with this command; however this is unnecessary as creation is automatic upon first reference elsewhere in the package. |
| *DELE[TE]: | Deletes "dead-wood" variables from the file. Deletes transparent variables (see text). |
| *GLOS[SARY]: | User can search the VARIABLE DICTIONARY and get data on all the SORTS which are current. |
| *STU[DENTS]: | The professor uses the command to handle any late student additions to his class file. |

### Table 7

### PROCESS DATA AND PRINTOUT RESULTS COMMANDS

| | |
|---|---|
| *AVER[AGE]: | Computes and prints out class averages and standard deviations of variables requested. (subsections of class calculations possible if used in conjunction with *INCLUDE-IF COMMAND) |
| *CORR[ELATION]: | Prints out a correlation matrix between any desired variables for entire class (or subset with use of *INCLUDE-IF command). |
| *TOTA[L]: | Computes class totals of variables requested (or for subset of class with use of *INCLUDE-IF command). |

### Table 8

### SECURITY COMMANDS

| | |
|---|---|
| *BATCH: | Passwords are not used in BATCH operation. SPARS will not run in BATCH unless the professor first opens BATCH up with the *BATCH command (via timesharing). Professor specifies how many jobs are to be let through in BATCH. |
| *LOCK: | Enables professor to enter or change his internal SPARS password. |

### Table 9

### PROCESS DATA AND STORE RESULTS COMMANDS#

| | |
|---|---|
| *ADD: | Adds up scores specified by the professor and stores the result where he specifies. |
| *ARIT[HMETIC]: | Multiplies, divides, and adds constant value to the value of a variable and stores result in variable specified. |
| *DROP[-WORST-SCORES]: | Same as *ADD except SPARS will drop the N-worst scores. |
| *GRAD[E]: | Cause SPARS to assign a grade to each student based on 11 specified threshold values of a specified raw score (i.e., grades on an absolute scale). |
| *NORM[ALIZE]: | This enables professor to remove and/or detect some of the grading inconsistencies between different labs or recitation sections. It produces a normalized score. |
| *RANK: | Determines the rank of each student for any variable and stores it in a specified variable. |
| *WAVG: | Weighted Average Command. Will compute the weighted average of scores and grades. |
| *ZGRA[DE]: | This command is equivalent to running *ZSCORE followed by *GRADE and then deleting the intermediate result. This computes a grade based on a curve specified by the professor using a raw score and placing the letter grade where specified. |
| *ZSCO[RE]: | Computes the ZSCORE (indicates how far a student is above or below class average in terms of standard deviation units) based on the raw score specified and stores result in a variable the professor specifies. |

#The operation of each of these commands can be caused to act on any subset of the class desired through the use of the *INCLUDE-IF command. The *TOSS command can also be used to affect the operation of these commands.

# Raising the Computer Consciousness at Skidmore College

**Margaret K. Brown**
*28 Crommelin Drive*
*Saratoga Springs, New York 12866*
*(518) 584-5279*

### The Founding of Skidmore

The Skidmore School of Arts was begun in 1903 by Lucy Skidmore Scribner. The latter's husband, publisher Charles Scribner, was of ill health and the Scribners came to Saratoga for rest and medical care. Lucy Scribner saw the need for educational opportunities for young women in the area and she organized classes in both the fine and practical arts. By 1911, the Board of Regents of the State of New York recognized the Skidmore School of Arts which grew to a four-year college and was granted a charter as Skidmore College in 1922.

The original campus was molded from restored Victorian mansions that graced the wide avenues in Saratoga Springs and from nearby studios remodeled into classrooms and laboratories. Growth of facilities and expanded curricula forced the college to acquire more and more buildings until there were over eighty individual structures, large and small, to maintain, police, and supervise. In 1963 the decision was made to acquire about 900 acres of wooded land in Saratoga Springs, but approximately two miles from the original campus, to build an entirely new home for Skidmore College. New academic programs were added, the college went on a new academic calendar and the decision was made in 1971 to become coeducational.

At the present time, the new campus is about 60 percent complete. Construction is continuing as rapidly as financing becomes available. At present, an academic building with classrooms, seminars and faculty offices is being built and is expected to be completed by spring, 1977. There are several important buildings not yet begun such as a gymnasium and art facilities. Additional living quarters will have to be added.

The curriculum is primarily devoted to liberal and fine arts. There are, in addition, a few professional areas such as nursing and education. At present there are just over 2,000 students, including 300 men. The proportion of men is increasing with each semester.

### Academic Computing at Skidmore

Skidmore does not expect to have a computer science major in the near future, but it is hoped that computing will serve a comfortable and useful role in the liberal arts environment at Skidmore. All administrative systems, with the small exceptions noted later, are contracted outside the college, and time sharing facilities on campus are devoted wholly to research and academic uses.

At present Skidmore has three lines to the Dartmouth Time-Sharing system. Our equipment includes:

Three DECWRITER II terminals
Three acoustic couplers
One Time-Sharing Peripheral Plotter (TSP)
One Digilog
One monitor on an easy-roll rack
One Student Response System

Responsibility for computer service resides in the organization called the Computer Applications Center which is housed under the mathematics department, but maintained on a separate budget. Its staff includes: one-third time director, one-half time secretary, and two student assistants working ten hours a week each.

There is a Computer Applications Center committee made up of eight faculty members who use the computing facilities, and two administrative representatives and the director as ex-officio members. This committee decides policy and direction for the Computer Applications Center and provides the director with suggestions and a summary of college reactions to the policies of the center. The director is a member of the mathematics department, works with the chairman of that department for the mutual support of the center and the department, and teaches mathematics courses for two-thirds of a regular load.

Administratively, the director: (1) represents the college at the task force on computers of the Mohawk-Hudson Consortium of Universities and Colleges and, in this capacity, keeps informed on administrative systems being developed at these computer centers. Informing the appropriate administrative office is the extent of the responsibility in this area; (2) maintains user accounts; (3) with the computer applications committee, endeavors to appraise our immediate and future academic computing needs and makes recommendations to the administration for direction and equipment. At present we are recommending the purchase of a mini-computer to replace all but one Dartmouth line; (4) maintains subscription journals and papers on computing, such as ***Computers in the Humanities, Creative Computing, Datamation,*** and ***Electronic News;*** and (5) attempts to raise the computer consciousness of the academic community.

The exciting part of the director's job is the last mentioned—to attempt to raise the computer consciousness of the academic community. This can and does take varied forms, some more successful than others. These efforts include: teaching courses, assisting individuals with their computer projects, meeting with groups of faculty members to suggest uses of computers in their field, and giving special talks.

A regular credit course in computer programming is given each semester. The fall semester is given in FORTRAN; BASIC is taught in the spring. This is a typical course in computing fundamentals taught everywhere. This course requires five semi-difficult programs (with structured flowcharts) and an individual project that the student must define, flowchart, and implement. We find that many students underestimate the difficulty of this course and drop out before completion. The curriculum committee is being petitioned to have this course qualify as satisfying one of the two laboratory requirements, but to date this action has not been taken.

Two no-credit, no-fee programming courses are taught each semester, usually for six sessions each. These are scheduled for various times during the day so as to meet as many schedules as possible. Noon hour is a popular time. We find some students, faculty, and staff attend, sometimes repeating the course more than once.

A no-credit, no-fee course is given on programs to operate the TSP plotter. TSP advertises that these programs can be written in BASIC or FORTRAN, but Dartmouth has not perfected the plotter routines in FORTRAN. The plotter course is usually given in three sessions and is primarily attended by students and faculty in the business and economic departments. This piece of equipment is relatively trouble-free and has been used successfully. Our only problem is that of pens that dry out rapidly when users forget to replace caps, and these pens are, in my opinion, absurdly expensive.

One course each semester is given also on some of the more popularly used utility programs in the DTSS Library, such as program computing: correlations for N observations on K variables, factional analysis of a design of up to fifty variables, and linear and multiple regression.

The above list covers the computing courses that are taught regularly, except that the mathematics department teaches statistics and numerical analysis which require computer programming. These are usually taken by mathematics majors only.

### SKIDLIB***

The Dartmouth Time-Sharing System has provision for establishing a library of programs that can be accessed from any user number. Typically, individual instructors will prepare programs on their own user numbers and then expect students from a class to access them, provide data, and execute them. Usually the instructor does not want his personal number to be made available to the community of students and so notifies the director as librarian of the sublibrary. She takes the action required to have these programs put in the library and adds a short description of them in the library catalog. In addition, data banks are established and put into the library for statistical analyses. Instructions on how to execute the programs must be included in the program before it will be added to the library. Persons needing help in executing the programs are encouraged to inquire at the Computer Applications Center.

Examples of such programs are (1) plotting routines, (2) programs that analyze chemistry data from experiments, (3) data banks on student surveys and faculty surveys, (4) data on campus growth and facilities, and (5) economic simulations.

### Departmental Usage of Computing Facilities

1. Chemistry classes collect data from laboratory experiments and run programs in SKIDLIB*** to analyze them.
2. Economic statistics classes write programs, usually in BASIC, implementing statistical routines, and plot data on TSP plotters.
3. Political science classes create data banks, e.g. on voting trends, and analyze them using statistical routines in system IMPRESS.
4. Mathematics students write programs for computer programming courses, and to implement numerical and statistical methods.

### Special Services

Although the Computer Applications Center has tried to stay away from involvement with administrative systems, it regularly undertakes two projects:

1. To produce labels for intra-college mailings for faculty council and provost's office.
2. To maintain information on students owing library fines and produce lists several times a year.

### Encouraging Signs That Computing is Permeating Other Departments

Recently two requests came from the English department for some help from the Computer Applications Center. First, interest was expressed in a program that would construct English sentences after the syntax for the sentence was specified and groups of words supplied. The set of all possible sentences so formed was typed as a result of execution of this lexical program. Insight was derived about the subtle interaction of syntax and semantics.

The second request came from the instructor of a course on the "Nature of Language." he asked if a talk on the nature of a programming language could be given to the class at the end of a month's time. After having taught programming language, concentrating on the syntax of individual statements, punctuation and the like, it was now necessary to take the long view and look at programming languages as a whole. Good reception was given to a discussion of how a computer language can be described in content free Backus Normal Form, and how it was possible to implement the languages so described by syntax driven compilers.

We have had requests from students that data from surveys be analyzed by computer. These requests are handled satisfactorily if the questionnaire is constructed with computer processing in mind. Sometimes the task of typing data into a file is more time-consuming than processing the reports by hand.

We had a request from the yearbook committee to produce the design for the yearbook cover. It was a picture of a Victorian building reminiscent of our old campus dorms. It was a tedious job, but our Computer Applications Center assistants received special mention in the yearbook.

Individuals often define specific projects and request programming assistance. The student assistants will write programs for faculty members if the request is reasonable. The director appraises and often designs the job before assigning it to an assistant so a subjective judgment of difficulty can be made. Faculty requests are infrequent. Students with independent projects usually are required to carry the burden of work themselves and so guidance rather than service is required. The bulk of the work handled by the student assistants is the on-the-spot assistance given to students with teletype or programming problems.

### The Student Response System

In 1970, Skidmore received an NSF grant for $70,000 in support of its proposal to enter into a cooperative research project with General Electric to determine whether or not the Student Response System (SRS) could be successfully used in the liberal arts curriculum of a college with small classes. The college built facilities for the system including a forty-seat classroom outfitted with movie, slide, and over-head projection in addition to the SRS. Each student desk has a set of buttons by means of which he/she can reply to factual, opinion or rating questions. From the console, an instructor can read meters indicating the percentage of students selecting each of the numbers and can vary lectures or discussions based on these responses. In addition, the responses can be reactions to projected materials.

The response from each desk can be recorded on a teletype wired into the system. When operating in group mode, every person responds to the same question during the given interval and the responses are recorded in desk order. When the system is operating in free mode, students can set the cog wheels to indicate what question is being answered and the desks are sensed in sequence repeatedly.

The SRS was our first big computer application requiring good computer service and good statistical analysis programs to process the responses. With the assistance of NSF, this project launched Skidmore into the computer age, almost out of the dark ages. Although the system is now seldom used, it did awaken much interest in computing at Skidmore and was the origin of our Computer Applications Center.

# Computer Assisted Instruction at Bennett College

**Nellouise D. Watkins**
*Computer Center*
*Bennett College*
*Greensboro, North Carolina 27420*

### Introduction

The middle of May 1976, marked the end of the first year of the formal use of computer assisted instruction at Bennett College. The first phase of the program involved the freshman class only because of the limitation of disc space—a problem we have now corrected.

A planning conference for the CAI program was held May 19-20, 1975, with a supplemental conference three days prior to the opening of school. The following decisions and procedures were operational by the opening of school:

1. The three discipline areas involved in the first phase would be English, mathematics, and reading.
2. Each different course would have control sections for comparative analysis of the effectiveness of CAI.
3. Syllabii in English and mathematics would be developed specifying course content. The object here was to assure uniformity of sections both experimental and control.
4. Diagnostic tests would be prepared to be administered for placement and evaluation purposes.
5. The CAI laboratory would be an integral part of the course with attendance by the students based on one or two days per week.
6. Weaker students would spend one day in the normal classroom setting and two in the CAI laboratory. Other students would have the reverse proportion.
7. The beginning instructional materials would be leased from Computer Curriculum Corporation until such time that our faculty could develop their own materials.
8. Grades would be weighted in the same proportions as classroom/terminal hours.
9. Certain topics which lend themselves most effectively to this media would be relegated as the responsibility of the terminal. The faculties would likewise have remaining topics as their prime domain.
10. The reading course would be managed and taught totally by the computer.

### Procedures

*Reading Program*

The reading course begins with a test on basic reading skills to ascertain an approximation of the reading level. The five reading components tested are: word attack, vocabulary, literal comprehension, interpretative comprehension, and work-study skills. When a student completes the diagnostic placement in 90 to 100 minutes, she is considered reading at the senior high school level and enters our regular reading program. Otherwise, the student is moved diagnostically through the skills until she is raised to the required level to enter the regular program.

The reading course, granting one semester hour of credit, is handled totally via computer. The student attends at her convenience as long as she clocks in forty-five hours per semester. The curriculum materials used are in social science, natural science and literature. Responses to the questions on these topics are recorded along with the amount of time spent at the terminal. The grade is determined by the amount of time clocked in. The decision to use this procedure of evaluation was based upon the rationale that the student should be neither penalized nor rewarded because of her previous experience. The program is designed to improve her reading skills regardless of initial level. The student has an option to 'test out' of the program when she can score ninety or above in each of the three discipline areas. She is allowed only one 'test out' option per semester.

### *English Program*

The English program yields three hours credit per semester. Placement levels are determined by SAT verbal scores and performance on an initial diagnostic test. Twelve common errors of grammatical usage form the base of the computer curriculum materials. The basic errors in English are: (1) run-on sentences, (2) sentence fragments, (3) incorrect principal parts of the verb, (4) confusion of adjective and adverb, (5) lack of agreement of pronoun and its antecedent, (6) lack of agreement of subject and verb, (7) incorrect case of pronouns, (8) vague or indefinite pronominal reference, (9) dangling elements, (10) misplaced modifiers, (11) errors in comparative forms of adjectives and adverbs, and (12) double negatives.

Students with SAT scores above 350 in the verbal section and/or less than six error types meet with an instructor two class meetings per week and one meeting at the terminal. All other students meet with the instructor once in the normal classroom and twice in the laboratory. Records are maintained by the computer for the entire semester on topics completed, scores for each session, and percentage of course completed. The final grade represents a weighted average of the instructor's grade for classroom performance and the grade determined by computer records. The allocation of the course content to be delegated to the instructor versus the computer was determined by the total English faculty prior to the beginning of school.

### *Mathematics Program*

The mathematics courses operate in the same manner as the English, except students in the highest level mathematics course write programs in the BASIC language during the latter portion of the course. The twelve common errors are: (1) arithmetic of counting numbers, (2) decimal arithmetic, (3) fractional arithmetic, (4) linear equations, (5) percentage, (6) word problems using percentage, (7) measurements, (8) algebraic expressions, (9) signed numbers, (10) quadratic equations, (11) graphing ordered pairs, and (12) factoring.

Each semester the program served 115 students in mathematics, 112 in English, and 130 students in reading. Classes were scheduled to accommodate fifteen students per hour with the teacher present for consultation and guidance. Each teacher secured a daily report of each student's progress. Periodic meetings of the faculties were held for evaluation and revision of procedures. In general, no major changes were made during the year; except for one main alteration of laboratory procedure in which the teacher gave a 10-minute capsule lecture on one of the gross errors.

## Evaluation

1. In English and mathematics where computer tutoring and drill were directed toward twelve gross errors in each area, both mid-year and end-of-the-year studies show the errors are not completely removed, but there was a significant reduction in the number of types of errors. For example, a student starting with all twelve of the types of errors seems consistently to be at the four to five level by the end of the year.

A comparison with the control group in the CAI sections also shows a reduction in error type, but the mean falls at seven to eight by the end of the year. This consistency was not evident in the control sections. The mid-year diagnostic test and the end-of-the-year test show error reductions are consistently the same errors.

2. In the computer reading courses no teacher was assigned and no classes scheduled. The student's grade was based solely upon time logged in at the terminal. Ninety-seven percent of the students had reached at least the beginning senior high school level by the end of the first semester.

The measurement instrument used in this case was a diagnostic placement tutoring drill program which would not allow the student to proceed until mastery of the given reading skill was demonstrated.

Only when the student scores 100 percent on each of the five skills is she told by the terminal to consult the teacher. The time required to reach this point ranged from 1 hour 47 minutes to in excess of 27 hours. At this point, the student is required to interact with the interpretation comprehension of reading in social science, literature, and natural science. Upon demonstration that these three booklets have been completed, the student can elect to test out of the program. This requires that the student score 87 or above in each of the three randomly selected tests from the above named discipline areas. By the end of the first semester, nineteen students had been able to test out. By the end of the year, this number had increased to twenty-nine.

## Analysis of Evaluation of Program

### *Reading Program*

1. Two and one-half percent of our freshmen in the computerized diagnostic placement test were reading at the senior high school level.
2. The time required for the completion of the program which determines senior high school reading level ranged from 70 to 120 minutes.
3. A mean time of fifteen hours was required for 33 percent of our students to be raised to the senior high school level.
4. Fifteen percent were able to qualify (by our criteria) as reading at the beginning college level within three months.

### *English-Mathematics Program*

Students were randomly assigned to control sections for each level of freshman English and mathematics. All students were administered initial tests covering specifically the twelve common errors of English and the twelve common errors of mathematics in August 1975. The same tests were given again in December 1975. The charts indicate the distribution of the number of 'common errors' at the beginning and end of the first semester for control and noncontrol groups.

**Mean Number of Errors**

*English*

| | Beginning | End | 1975 Class Sample |
|---|---|---|---|
| Control | 8 | 7 | 9 |
| CAI | 8 | 5 | |

*Mathematics*

| | Beginning | End | 1975 Class Sample |
|---|---|---|---|
| Control | 9 | 10 | 9 |
| CAI | 10 | 7 | |

Error Reductions

*English*

| | % reduction |
|---|---|
| Control | 55 |
| CAI | 73 |

*Mathematics*

| | % reduction |
|---|---|
| Control | 60 |
| CAI | 66 |

## Reading Statistics

*CAI:*

***2.5 percent of the students were reading at the senior high school level at the beginning.***

***15 percent of the students were reading at the beginning high school level within three months.***

***3 percent of the students failed to reach the high school level. It required a mean time of 11 hours and 41 minutes for students to reach the beginning high school level.***

*Control:*

The same diagnostic test that was administered to 130 students at the beginning of the CAI reading course was given to twenty control reading students at the end of the school year.

The statistics were as follows:

No student in the control group was diagnosed as reading at the beginning high school level. This was based on less than two hours completion time.

One student was at 81 percent completion in two hours. Based on a proportional time formula, at the completion of a year of reading instruction the mean time required for control students to reach beginning high school level would be nine and a half hours.

Mean time required for reaching the same level using the computer without a teacher and no previous reading instruction was eleven and a half hours.

## Analysis of Results

*General:*

PROS

1. High motivation of the hardware and software.
2. What teacher would put in 1,050 student hours per week?

CONS

1. Computers have "down" time.
2. Natural human reaction to fear of replacement by a machine.

*Reading:*

PROS

1. Program contributed in helping the student develop a habit of reading.
2. Increased the awareness of need for improving reading skills.
3. Aided the student in increasing power of concentration.
4. The selection of reading materials in the content areas—social science, natural science, and literature—gives students facts and general concepts for courses in which they are currently or will be enrolled.

CONS

1. We have a weak rationale for assigning letter graders—based on a time factor alone.
2. Additional reading materials are needed to avoid repetition before competency level is reached.
3. Weakness of depending on the maturity of the student to avoid procrastination to a point of no return.

*English-mathematics programs:*

PROS

1. Provides for drill on a one-to-one basis.
2. Provides interaction, tutoring and corrections without embarrassment of teacher or other students.
3. Allows the student to self pace her learning.

CONS

1. Reluctance of faculty and students to accept change.
2. Infringement on academic freedom of faculty; loss of ego.

## Future Projections

1. Our faculty began in the summer, 1976 to write modules and/or entire courses in the general education block to be put on the computer.
2. Provisions are being made for honor students to complete courses under an accelerated option contract for independent study with the terminals playing a major role.
3. As of the fall term, 1976, we began developing 'test banks' for most of the courses offered so that students may use this source in preparation for final examinations.
4. Random selection from the test banks will provide individual tests to be administered and scored by the computer.
5. We plan to make available the freshman English course to superior local high school students during their senior year.
6. We plan to use the college computer facilities in academic reinforcement in mathematics for elementary school children. The goal here is to help children as well as improve our community image.

# Interactive Graphics with a Microprocessor Terminal[1]

**John H. Karl**
*Department of Physics/Astronomy*
*University of Wisconsin-Oshkosh*
*Oshkosh, Wisconsin 54901*
*(414) 424-4431*

The University of Wisconsin-Oshkosh has been using interactive computing in its undergraduate curriculum since 1969.[2] However, due to the modest budget available for interactive computing, this component of the curriculum has been restricted to limited usage in a few courses in the departments of chemistry, physics, mathematics, computer science, psychology, and the school of business. The recent purchase of three Tektronix 4051 graphic system terminals from local and NSF funds under the CAUSE program has sparked much interest from faculty in other departments, such as geology, medical technology, and economics as well as from the ones already using interactive computing.

Of course, the added dimension of graphics to the interactive environment is largely responsible for this increased interest among faculty who are now using or are potential users of interactive computing in their instruction.

A second, and possibly more important, characteristic of the 4051 micro-processor system of concern to the faculty is its ability to operate in a stand alone mode as a BASIC interpretor with 32K (eight bit bytes) of RAM and a 300K tape cartridge. The graphic 4051 BASIC is sufficiently powerful for use in CAI applications where self-documented, practice-drill lessons require multilevel branching, character string manipulation, and large amounts of storage. Faculty can thus embark on the development of computer programs and student courseware with confidence that the system will not be later changed by a committee governing some remote host computer facility. This last consideration is becoming increasingly important in the development of large PSI courses which depend on a reliable current courseware for their smooth implementation.

As a component of UW-O's CAUSE project, eight faculty in astronomy, chemistry, geology, medical technology, and physics are developing 4051 software and appropriate courseware for use in more than twelve courses in these fields. Although their instructional units written for the 4051 GRAPHICS BASIC do include some interesting aspects in connection with the 4051 magnetic tape files, joystick operation, stand alone operation, and storage screen limitations, nothing particularly innovative is contained in them. Thus, the physics instructional units are probably exemplified by lessons like Takat's[3] in quantum mechanics, Beeman and Wolf's[4] in upper-division physics courses, and Merrill's book on computers in physics.[5] In any case, the complete output of the computer component of the UW-O CAUSE project will be made available on request in a unified documented format for 4051 programs with student courseware and thus need not be reported here.[6]

What may be of interest is the use of this relatively new device as an intelligent graphics terminal. The 4051 itself is computationally rather slow (about three msec add time) compared to common time shared systems which utilize large host computers. Thus it is important to report on using the 4051 to graph locally while doing the large computations on a host computer.

### The 4051 as an Intelligent Terminal

The 4051 can communicate to a host computer via an RS-232 ASCII interface. It is possible to exchange data or programs between the host computer and the 4051 memory or magnetic tape unit. For example, a BASIC program can be written, edited, and even debugged on the 4051 off line and then transmitted to the host for fast execution. This feature saves considerably on connect time, and thus also increases performance of the time-shared system. If a graphics package is not available on the host computer, data can be sent to the 4051 which then uses its BASIC language graphics to do the plotting locally. In this mode of operation a slight sacrifice is made in transmission time because the data must be sent in ASCII digit characters and converted to floating point in the 4051 terminal.

In order to explore this mode of use of the 4051 graphic system, an electrostatic field plotting routine was selected from one of the lessons in introductory physics for use as a "benchmark" program. The self-documentation and student interactive portions were removed to produce a demonstration plot of a linear electrical quadrapole charge distribution. The program was then loaded into the IBM-370/145 CALL-OS time shared system under BASIC. A short program was written in BASIC for the 4051 which "runs" the time shared system, inputs data from the remote computations, and graphs it.

## The Programs

An annotated listing of the quadrapole computation routine is shown in Figure 1. It uses the half step method for computing both electric field lines and equipotential lines as described by Merrill.[5] In lines 250 to 500 and lines 380 to 420 it outputs on the telecommunications line a flag for E-field lines or equipotential lines, the number of coordinate pairs for each contour, and then the coordinates themselves.

The starting points for each contour, which are usually determined by the student positioning the joystick, are generated internally for this demonstration exercise.

**Figure 1**

**An Annotated Listing of the Computational Program Which Resides in the IBM-370/145 CALL-OS Time Sharing System**

```
100 N=3
110 DIM U(400), V(400)                              X & Y plotting points
120 DIM Q(3),X(3),Y(3),B(14)                        Charges and their locations
140 DATA -1,2,-1
150 DATA 50,50,50
160 DATA 52,54,56,58,60,62,64,56,68,70,72,28,30,32
170 MAT READ Q,X,Y,B
175:##.#
180 R9=4
190 E1=1
200 FOR K1=1 TO 24                                  Starting points for E-lines
210 A1=15*K1*3.14159/180
220 X0=R9*COS(A1)+50
230 Y0=R9*SIN(A1)+50
240 GOSUB 450
250 PRINT "E"                                       Output over time shared system
260 PRINT L                                          lines 250 to 300 & 380 to 420
265 PRINT X0
266 PRINT Y0
270 FOR K=1 TO L
280 PRINT USING 175,U(K)
290 PRINT USING 175,V(K)
300 NEXT K
310 NEXT K1
320 E1=-1
```

**Figure 1**

**An Annotated Listing of the Computational Program Which Resides in the IBM-370/145 CALL-OS Time Sharing System (continued)**

```
330 FOR M=1 TO 14                                  Starting points for V-lines
340 X0=B(M)
350 Y0=50
360 GOSUB 450
380 PRINT "V"
390 PRINT L
395 PRINT X0
396 PRINT Y0
397 FOR K=1 TO L
400 PRINT USING 175,U(K)
410 PRINT USING 175,V(K)
420 NEXT K
430 NEXT M
440 GO TO 850
450 REM *** START CONTOUR ***
460 L=0
470 X1=X0
480 Y1=Y0
490 F1=0
500 F2=0
510 F3=0
520 F4=0
530 FOR I=1 TO N                                  Sum fields of all point charges
540 X3=X1+F1*0.5-X(I)
550 Y3=Y1+F2*0.5-Y(I)
560 R=SQR(X3*X3+Y3*Y3)
570 R3=1/(R*R*R)
580 F3=F3+Q(I)*X3*R3                              Sum of Ex
590 F4=F4+Q(I)*Y3*R3                              Sum of Ey
600 NEXT I
610 E=1/SQR(F3*F3+F4*F4)                          Absolute value of E
620 IF E1=-1 THEN 710
630 F1=F3*E                                       Increments of X for E
640 F2=F4*E                                       Increments of Y for E
650 X1=X1+F1                                      New X
660 Y1=Y1+F2                                      New Y
670 L=L+1                                         Length of E contour
680 U(L)=X1                                       Store X coordinate for E contour
690 V(L)=Y1                                       Store Y coordinate for E contour
700 GO TO 780
710 F1=F4*E                                       Increment of X for V
720 F2=-F3*E                                      Increment of X for V
730 X1=X1+F1                                      New X
740 Y1=Y1+F2                                      New Y
750 L=L+1                                         Length of V contour
760 U(L)=X1                                       Store X coordinate for V contour
770 V(L)=Y1                                       Store Y coordinate for V contour
780 IF ABS(X1-X0)+ABS(Y1-Y0)<0.9 THEN 840         Test for close to starting point
790 FOR J=1 TO N
800 IF ABS(X1-X(J))+ABS(Y1-Y(J))<1.5 THEN 840     Test for close to charges
810 NEXT J
820 IF X1>90 THEN 840                             Test for off page
825 IF Y1>90 THEN 840                             Test for off page
830 GO TO 510
840 RETURN
850 END
```

The BASIC program in Figure 2 is a little less ordinary. The @ 40 in the I/O commands is the primary address of the data communications interface. Thus line 140 sends a "run" command to the time shared system.

**Figure 2**

**The BASIC Program Which Does Scaling, Converting and Plotting Resides in the 4051.**

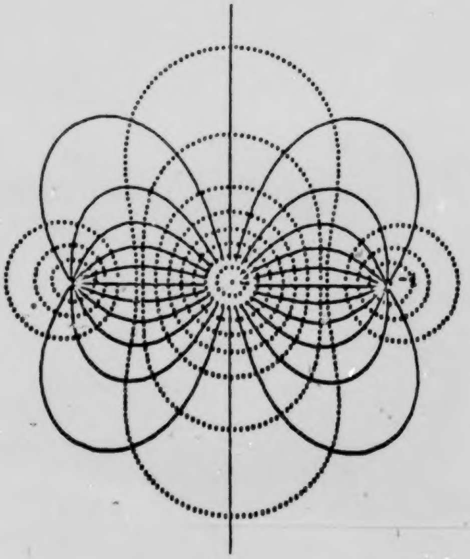
```
100 PAGE
110 INIT
120 VIEWPORT 15,115,0,100
130 WINDOW 0,100,0,100
140 PRINT @40:"RUN"
150 INPUT @40:A$
160 INPUT @40:B$
170 INPUT @40:E$
180 INPUT @40:L$
190 INPUT @40:U$
200 INPUT @40:V$
210 MOVE VAL(U$),VAL(V$)
220 FOR I=1 TO VAL(L$)
230 INPUT @40:X$
240 INPUT @40:Y$
250 IF E$="V" THEN 280
260 DRAW VAL(X$),VAL(Y$)
270 GO TO 300
280 MOVE VAL(X$),VAL(Y$)
290 PRINT "."
300 NEXT I
310 GO TO 170
```

The INPUT commands read data in from the program which is running on the host computer. The first two character strings, A$ and B$, are run "labels" from the time shared system which are of no interest and are thus "swallowed" by the 4051 program. The E-V flag, contour length, L, and the plotting coordinates U, V, X, and Y are then read in and converted to their floating point numbers with the VAL function. The final output of the program is shown in Figure 3.

This three-charge quadrapole was selected as the benchmark problem because its total execution time is the same using the fast time shared system as using the 4051 stand alone. (Execution time for the entire plot shown in Figure 3 is about fourteen minutes.) The difference is that with the host computer doing all the computations, essentially all the time is spent in data transmission (300 baud). But with the 4051 doing all the computations, all the time is spent in computing. Thus, any program with more computations and/or less I/O than this, would be faster on the host computer. Any program with more I/O, and/or less computing, would be faster on the 4051 stand alone. For example, if it was desired to plot the field due to six changes instead of three, no significant increase in time would occur using the host computer providing the number on contours remained the same.

**Figure 3**

**Electric (Solid) and Equipotential (dotted) Lines for a Linear Quadrapole Produced by Computations on a IBM 370/145 Time Shared System and Plotted by a Tektronix 4051 Graphic Terminal.**



## Conclusions

The 4051 graphic system terminal is a versatile machine. It can act like a pure ASCII graphic terminal, it can do stand alone graphics in BASIC, or as seen here, both modes can be combined. With the increasing power and lowering cost of microprocessors, systems like the 4051 are going to find increasing importance in the use of computers in the curriculum. Their stand alone capabilities are particularly important for CAI applications that require considerable I/0 coupled with only small computations. For computations that require a large host computer, graphics can still be done in a high level language without any particular additional requirement at the host computer site.

## Notes

1. This work is supported, in part, by the National Science Foundation Grant 76-16644.

2. Karl, John H. "Computerized Experiments Using an A/D Converter," proceedings of the CCUC, 1971.

3. Takats, M. C. "Teaching Elementary Concepts in Quantum Mechanics with Computer Graphics," proceedings of the CCUC, 1976.

4. Beeman, D. and Wolf, R. P. "Computer Graphics for Upper-Division Physics Courses," proceedings of the CCUC, 1975.

5. Merrill, John R. *Using Computers in Physics*, Houghton Mifflin, Boston, 1976.

6. Copies may be requested from the author.

# Use of Computer Graphics in the Physics Laboratory at Washtenaw Community College

**Dwight D. Hinds[1] and George H. Kapp**
*Washtenaw Community College*
*Ann Arbor, Michigan 48106*
*(313) 971-6300*

## Introduction

"Datacrusher" is an interactive computer program that enables a student with no prior instruction in computer programming to sit down in front of a graphics terminal and obtain useful graphic information from laboratory data. The package was developed and written by George Kapp; it was implemented in the physics program at Washtenaw Community College by Dwight Hinds. It was created for beginning college physics students at Washtenaw and has been used by both our freshman students and our sophomore calculus-level physics students. It enables students to find such information as a best fit equation for their data, the slope of a curve at a point, and the area under a curve.

At the time of inception, it was clear that in order to be useful for students, certain criteria must be met:

1. The typing required to direct the package to perform a task must be kept to a bare minimum.

2. The amount of instruction needed for completing a graphing session must be minimized.

3. Meaningful prompts must be issued in order to achieve the proper response from the student.

4. The package must be "bullet proof," i.e., input errors should not cause it to crash.

It was also clear that to be a useful teaching tool from the instructor's viewpoint, the graphing package must be relatively inexpensive to operate, must be versatile, and must be easily modified. To accomplish these goals, extensive use of graphical input (pointing at objects on the screen) was necessary.

## Description

All commands available in the package are listed in various display "menus." The selection of any menu item involves merely pointing to that entry. The various commands displayed in the menu are grouped into sections of common usage to assist the student in location of options.

Three different modes of data entry are provided the student: (1) typing in *x*, *y* coordinate pairs, (2) entering a mathematical equation, and (3) drawing the curve on a digitizing tablet. This data is automatically scaled to the proper size, but the scale may be changed later if desired. An editor is available so that a student may change, delete, or add new coordinates. Facility for storage and retrieval of data sets on permanent file is available. This enables a student to terminate a session and later resume work without having to re-enter data.

Datacrusher has a number of special display parameter options. They allow the student to change the number of significant figures displayed, to change the region of the graph displayed (the window), and if there are a number of defined data sets, to determine which ones are to be displayed.

Intersection point, slope, area under a curve, and length of a line can be obtained. All of the information returned to the student is computed from the graph definition and not from a mathematical technique. The student could duplicate this information from a similar paper copy using a ruler. This is done so that the student does not feel the program is magical.

The curve-fitting routine is a very popular feature of Datacrusher. The student's data set and a number of equations commonly used in physics are initially displayed. The student may then select an equation for fit and Datacrusher will, via least squares, compute a fit for his data. The "best fit" curve will be overlayed on the original data along with the "best fit" equation. The fit curve may be substituted for the original data for further analysis. The student may, of course, try a number of different equations and view the results.

Although the need for hardcopy has not been in great demand thus far, the provision exists. A request for a copy of anything on the screen may be sent to a Calcomp, three-color plotter at any time during the run. The size of this plot is variable (from 3 to 27 inches). Each curve may be assigned a plotting color.

A twelve-page manual is available to students. It describes each feature of the program, the capabilities and the limitations, and includes sketches of typical output.

### Method

The graphing package was introduced in the fall semester of 1976 in two sections of General Physics III, a freshman algebra-level physics course. At the same time a number of physics courses, including General Physics III, changed over to an "open laboratory" format. To give the reader a sense of the Datacrusher user's environment, the main features of the open lab are summarized below:

1. The staff (four instructors, a laboratory coach, and a student assistant) provide coverage for five first-year physics courses (about thirty-five to forty hours per week).
2. Laboratory consists of seven two-week units, each unit being comprised of about seven or eight short experiments.
3. Equipment is placed at numbered stations along with one or two page instruction sheets. It stays up for two weeks. Laboratory instructors assist whenever requested.
4. Student can use any of the available hours to come in and work. They can stay as long as they like or come as often as they like. This particular mode of operation is advantageous in that with only two graphics terminals available (one of them used for other computing purposes), the period of use can be spread out. However, not all of the instructors were fully familiar with the graphing package, so students tended to come in when Kapp and Hinds were on duty.

The graphing package was introduced on an optional basis to interested students in General Physics III during the second week of the course. Sixteen of thirty-six students elected to use it. Each student was given an ID number and password and a copy of the user's manual. A one-half hour demonstration was arranged for the following week. Students were asked to sign on and access Datacrusher and report back their account balances. This gave the users some familiarity with the program and with the computer terminal.

Three problem sets were assigned during the semester, each one containing three different problems. Students could do as few or as many of these nine computer analysis problems as they desired, with their laboratory work being reduced accordingly. The problems pertained to previously done or current laboratory experiments. The input data were actually collected in the laboratory. A list of the problems with some descriptive information is shown in Table 1.

Prior to the use of Datacrusher in fall semester of 1976, the prototype version of Datacrusher was tested in a number of courses by the authors and a colleague, Gerald Rees. The following four applications seem to be useful and we plan to use them again with Datacrusher:

1. Photoelectric effect experiment—Planck's constant found from best linear fit on stopping voltage vs. frequency data.
2. Radioactive decay experiment—half-life of a radioactive isotope found from best exponential fit on activity vs. time data.
3. For hypothesized data, Planck's radiation law curve was plotted and the maximum intensity wavelength was determined.

**Table 1**

| *Title of Experiment* | *Data Entry* | *Functions* | *Result* |
|---|---|---|---|
| 1. Projectile motion | Strobed photograph via tablet | Parametric separation:<br>Linear fit;<br>Parabolic fit | Components of motion equations |
| 2. Falling object | x, y pairs | Linear fit; slope | Acceleration of gravity |
| 3. Energy in a bow | x, y pairs | Area under curve | Work done |
| 4. Spring constant | x, y pairs | Linear fit; slope | Spring constant |
| 5. Boyle's law | x, y pairs | Area under curve | Work of compression |
| 6. Elastic collision | Simultaneous equations | Overlay;<br>Intersection point | Solution set |
| 7. Charles' law | x, y pairs | Linear fit;<br>Intersection point | Absolute zero |
| 8. Cooling curve | x, y pairs | Exponential fit;<br>Other curve fits | Best fit equation |
| 9. Simple pendulum | x, y pairs | Power fit | Acceleration of gravity |

4. Using data for a diesel engine, the work accomplished per cycle of the engine was found.

### Results

Based on one semester's observation, personal discussion with student users, records kept by Datacrusher on its usage, and the results of a computer use questionnaire, the graphing package appears to have been well received. Since Datacrusher's implementation occurred simultaneously with the change to the "open lab" system, the overall positive reaction is very encouraging.

Sixteen out of a possible thirty-five students used the graphing package in the General Physics III sections taught by Hinds. The total cost for computer time was $201.67. Students signed on 101 times and were online 2,449 minutes. The average student used Datacrusher 6.3 times during the semester, spent $12.60 in computer time, averaged 24 minutes per session, used the Tektronic 4010 graphics terminal 88 percent of the time (DTC—300 hardcopy terminal was used 12 percent), and completed about 6 of the 9 assigned problems.

Students generally preferred the Tektronic 4010 graphic terminal to the DTC—300 hardcopy terminal due to its greater speed and the availability of graphic input. The one-half hour introductory session was rated adequate and the user's manual was considered very helpful. However, a number of students indicated that experience in using the terminal and the program is the only real teacher. The only problem in using the program seemed to be difficulty in maneuvering between sections of the menu. This was generally encountered after the student had made an error in usage.

Students were about evenly divided on the issue of whether an assistant should be physically present during a run. Some favored direct assistance, while others leaned toward the learning-by-doing approach. By a large majority, students favored computer-related labs over non-computer-use ones.

They clearly preferred getting laboratory data and then going to the computer over the simulation approach. All students felt that the use of Datacrusher should be optional and not required of all students.

The major advantages of the graphing package were stated by students as:

1. It is fun and interesting.
2. The exposure to the computer and to computer graphics is valuable.
3. It provides instant graphing as results are obtained quickly and easily.
4. It helps in the organization and analysis of laboratory data.

The major weaknesses were listed as:

1. Crowded conditions. Sometimes there were more potential users than terminals available.

2. Delay time and transmission errors occasionally occurred during runs.

3. Time spent on the computer reduces the time available for laboratory work.

4. On occasion there was no assistant immediately available.

## Conclusion

We feel that given the complexity of the graphing package, the unfamiliarity of computer terminals to many students, and the start-up problems associated with an "open lab" format, the results of datacrusher usage in General Physics III are quite positive. Student enthusiasm and interest were high. The quality of students' work was surprisingly good. The computer time cost figures are reasonable. In addition, students who use Datacrusher in later physics courses at Washtenaw Community College will not need the initial training phase. The benefit-to-cost ratio will increase for these students.

The major advantages of Datacrusher as listed by students tended to be learning outcomes (how to use a computer, enjoyment, fast graphical results, etc.). The major disadvantages were those related to computer use (crowded terminals, delay in transmission, lack of direct assistance, etc.) rather than to a lack of learning.

To maximize the advantages and minimize these disadvantages, we plan the following revisions in method and material:

1. A follow-up Datacrusher use demonstration will be offered a few weeks after the initial session.

2. More direct assistance to the student while on-line will be provided at first with less later on.

3. More detail in problem handouts will be provided at first with less later on.

4. An error analysis handout will be developed to assist students on how to proceed when common mistakes occur.

5. A handout showing a "step-by-step solution of a specific problem using Datacrusher" will be developed.

## Implementation and Hardware Overview

The graphing package, Datacrusher, was written with a combination of FORTRAN and assembly language. The actual driving routines, the analysis routines, and the data structure are in FORTRAN. The graphics support is in assembly language. Extensive use of system subroutines for file handling, device recognition, and keyboard input error recovery was also incorporated. The data structure uses dynamic memory acquisition rather than FORTRAN arrays in order to keep costs down. Parts of the program also call the system FORTRAN compiler as a subroutine to compile user equations; the object file is then linked to Datacrusher during execution.

The environment in which Datacrusher runs is as follows: (1) University of Michigan Terminal System (MTS), (2) AMDAHL 470 computer, (3) Tektronic 4010 graphic storage terminal with a 4953 Tektronic graphic input tablet, and (4) Data Terminal Corporation (DTC) model 300 terminal with plotting option.

## Note

Correspondence with regard to the contents of this paper should be directed to Professor Hinds.

# Non-Interactive Graphics Techniques in Mathematics Education

**Kenneth Joy**
**James R. Warner**[1]
**Richard Yuskaitis**
*Graphics Education Research Group*
*Colorado University Computing Center*
*3645 Marine Street*
*Boulder, Colorado 80309*
*(303) 492-8131*

## Abstract

Computer-generated imagery can accurately display two- and three-dimensional illustrations from mathematical models. High precision images are economically developed and displayed in a batch environment using a compositional modeling system. Classroom visual aids (e.g., large prints, slides, transparencies) are obtained from the computer image.

The production methodology for a specific series of illustrations used in introductory calculus courses is discussed.

*Keywords*

Computer Graphics, Audio/Visual Aids, Mathematics Education.

*Computing Reviews Category:* 8.2

## Introduction

A need exists for new *visual* approaches to mathematics education. Learning theories suggest various cognitive styles of learning, and mathematics is one discipline in which the visualization process is extremely important. However, very little effort has been applied to mathematics education problems involving innovative visual aids for college mathematics. Few attempts have been made to enhance the interpretation, understanding, and retention of the complex, often interrelated, concepts a student must visualize in learning mathematics.

In the classroom, certain simple illustrations can often be sketched with sufficient precision for student understanding. Mechanical aids (e.g., compass, ruler, protractor) are usually adequate for creating two-dimensional visuals. Many concepts, however, in undergraduate mathematics curricula cannot be sketched with enough precision to enhance the learning process. The students' understanding must evolve solely from abstract symbolism without an adequate visual model.

While text book illustrations reflect precise draftsmanship, many artist renderings, especially three-dimensional models, are both inexact or incomplete, failing to illustrate mathematical derivation.

The computing machine is the obvious means for precisely simulating mathematical concepts and problems. Computer graphics techniques have the potential for mapping these abstract phenomena into accurate visual models.

Two approaches exist for utilizing the computer graphics medium for mathematical illustration. The first approach involves batch computing with graphics output to a high resolution display device. This passive use of graphics implies a deliberate modeling process geared toward optimizing the educational effect of the computer-generated imagery. The combined factors of batch computing, passive graphics, and detailed models, imply a moderate time span between image inception and image generation.

The second approach is the use of interactive graphics devices for generating real-time images directly in the instructional setting. Real-time mathematical imagery is usually a single-purpose program or a subset of a CAI system. In a classroom setting, the instructor can systematically step through a series of illustrations dynamically generated on an interactive display viewed by the class. Outside of class, students may individually step through canned tutorial programs on the current topic. Either use of real-time graphics lends an aura of magic to the instructional process and, as a result, often expands the students' interest.

While interactive graphics is an intriguing teaching tool, several limitations should be noted. The educational content of CAI graphics is constrained by the programming talents of the lesson coordinator. While higher level CAI systems ease the programming burdens, the student or instructor rarely can explore material that is tangential to the central topic.

Unless canned images are being copied from peripheral storage to the display, significant computing time may be required to compute and display the image. This is compounded if students are viewing the same images in independent sessions.

Real-time graphics terminals are expensive. Moreover, the diversity of terminals and lack of a graphics software standard hamper the exchange of programs among educational institutions.

The Graphics Education Research Group (GERG) of the Colorado University Computing Center was formed in the spring of 1976 to investigate and develop economically viable computer visualization techniques for use in the educational process. Recognizing the relative importance of existing mathematics illustrations, our initial project was to design and develop a cogent series of illustrations to act as visual aids in teaching college calculus. This has proven an ideal topic since many concepts of the calculus require illustrations that are virtually impossible to sketch. Textbook visuals are often vague or imprecise. Few visual aids have been produced, other than some super-8mm film loops and 16mm films.

We determined that economic viability could be achieved by carefully designing and generating illustrations with a universal appeal, rather than selected images relevant to only a single classroom environment. We reasoned that a high development cost could be easily regained by a broad distribution of the illustrations.

The precision requirements and developmental time span, coupled with the general imbalance of interactive graphics devices among educational institutions, immediately ruled out real-time graphics as a medium for teaching calculus. The restrictions in the image content and limited distribution outweigh the merits of dynamic interaction.

The remainder of this paper details the developmental criteria and general methodology for generating the calculus series illustrations. Selected illustrations from the calculus series are detailed in the Appendix.

## The Calculus Series Illustrations

The calculus series is a selected set of visual aids for use in a lecture-recitation classroom environment. In most settings, the instructor lectures two to three hours per week. Problem solving takes another one to two hours per week. The instructor relies heavily upon the blackboard with references to the textbook. Illustrations are limited by the instructor's draftsmanship and any pictures in the text.

Our experience, both in the class and in front of the class, has shown that few instructors can accurately sketch mathematical imagery. While the text is normally adequate for two-dimensional illustrations, few texts supply accurate models of the more difficult three-dimensional material. Students must rely totally upon the abstract concepts with no visual models. Many potentially competent students are unable to grasp the theory without illustrations to support the concepts.

We determined that supplemental visual aids illustrating these mathematical abstractions would greatly enhance the instructor's presentation.

While computer graphics techniques are the only viable means for generating precise illustrations, it is economically infeasible to have computer graphics hardware in every classroom. All educational institutions have portable audio-visual equipment (e.g., slide projectors, overhead projectors, 16mm projectors) for classroom use. Instructional materials for mathematics should take advantage of these universally available aids. We subsequently chose to generate the mathematical visuals on high resolution graphics devices, and then convert the computer generated imagery to conventional visual media using photographic techniques.

The calculus series consists of forty-three illustrations depicting two- and three-dimensional mathematical problems and concepts. Each illustration was programmed as a mathematical model using a compositional graphics system at the Colorado University Computing Center. The imagery was previewed and edited on an interactive graphics terminal. Final images were produced on the Information International FR-80 microfilm recorder. The 35mm microfilm was then photographically transferred to create 2 x 2 inch slides and 8 x 10 inch transparencies. This allowed the visual to be used in the classroom with conventional audio-visual equipment (i.e., slide projector, overhead projector).

The next section details the computer techniques used in generating the calculus series.

## Computer Graphics Techniques

While this discussion is primarily concerned with computer graphics techniques in mathematics, the procedures followed in image generation may be extended to any scientific discipline requiring precision simulation.

### *Methodology*

The process of creating computer generated educational imagery may be viewed as a staged process with interrelated phases of development and analysis. Each of these production stages is now reviewed.

### *Concept Clarification*

The first, and perhaps most critical, phase is the clarification of the concept to be illustrated. Is it a single concept or closely related with other areas? What type of images will best illustrate the concept? Why are computer techniques required or desirable?

With a well defined profile of the educational requirements and a skeletal outline of the graphics needs, the author/instructor should meet with the computer analyst and audio/visual coordinator to establish the viability of concept illustration.

### *Viability Analysis*

Although computer techniques may be highly desirable in visualizing a concept, the initial graphics requirements may exceed the available graphics facilities. It is at this stage that the computer analyst and audio-visual coordinator must review these graphics needs with the instructor. Sometimes, the visuals may be simulated exactly as outlined by the instructor. This is often true of two-dimensional imagery. More often, certain revisions in the initial design will yield a substantial savings in computing expense without sacrificing the content. The computer analyst must later merge the instructor's intent with the capabilities of available graphics simulation software in developing a computer model of the imagery.

Occasionally, the conceptual imagery cannot be easily amended for computer graphics simulation. It may be beyond the capabilities of either the available hardware (e.g., resolution requirements) or the software (e.g., tonal rendering, hidden-line processing, etc.). Often, though the imagery can be simulated, the computing cost, both in program development and CPU charges, outweighs the potential educational gain from the imagery.

### *The Computer Model*

With a detailed outline of both the image requirements and the mechanism for simulation, the actual computer model is built within the constraints of the available hardware and software. Ideally, the imagery is defined parametrically such that simple prototype models may be developed with a minimal computing overhead (e.g., without hidden line processing, without tonal renderings). These basic models may be reviewed and revised in collaboration with the instructor.

When the prototype meets the illustration requirements, the final imagery may be generated. Final output is normally to a high resolution graphics device, amenable to any post processing technique. Depending on the computing cost, several views of the final model may be generated. The instructor may then choose those images that most clearly define the concept.

### *Graphics Hardware*

Several graphics configurations may be acceptable, depending on the resolution requirements of the imagery. The only critical component is some type of vector plotting device. This may be a drum or a flatbed plotter, storage tube or refreshed CRT, or a high-resolution microfilm recorder.

While images may be photographed directly off a plotter or CRT, the microfilm recorder is the ideal device for generating high-resolution, registered imagery. The 35mm film may be shot as either a positive (black lines on a clear background) or reversal (clear on black) image, and is easily post-processed into slides, prints, or transparencies. Image previewing is best done on an interactive CRT. The analysis and revision stage of development can then be done online, with the final imagery generated later to the microfilm recorder.

In the development of the calculus series at Colorado University, the prototype models were reviewed on both the Tektronix 4015 storage terminal and the DD-80 microfilm recorder. The DD-80 is a *low*-resolution microfilm recorder that is good for previewing, but is incapable of generating the precision required in the three-dimensional models. Final imagery was shot on a high-resolution microfilm recorder (III FR-80) at a computing installation in California.

### *Software*

The simulations generated for the calculus series were generated using the MIDAS compositional modeling system [2,3]. MIDAS was developed at Colorado University as an animation package in 1974, and has evolved into a general graphics simulation system based upon hierarchical modeling. The MIDAS package is a library of FORTRAN-callable subroutines for defining, transforming, and viewing two- and three-dimensional imagery. The routines are written in FORTRAN and, wherever possible, conform to the PFORT [1] standards of portability. The graphics generation routines of MIDAS are designed to be compatible with any graphics device with vector drawing capabilities. Hence, the same simulation program may be used for interactive previewing on a storage tube and final image generation on a microfilm recorder.

Several software features have proven particularly desirable for modeling mathematical imagery. Picture components are defined in either a two- or three-dimensional cartesian coordinate system. The units of the coordinate system are arbitrary. Mapping from this coordinate system to any graphic device is automatically handled in the MIDAS software. Variable viewing parameters define a virtual camera allowing any focal point, camera position, picture plane, and angle of vision.

Each picture component has a series of attributes indicating color, intensity, and point configuration (e.g., polygonal or mesh). Hidden surface processing enhances three-dimensional displays. Polygonal components may be shaded using parallel vectors of arbitrary orientation, spacing, and intensity.

### *Post Processing*

The 35mm reversal microfilm image provides the greatest flexibility for conversion to conventional visual media. The reversal image can act as a high contrast negative and be printed directly onto photographic paper or on 8 x 10 inch positive film for use as a transparency. Lower quality transparencies may be obtained using Xerox or Thermofax transparency production process.

By using transparencies with the overhead projector, multiple overlay images can systematically build a compositional model. Color may be easily and inexpensively added by applying pressure-sensitive tints to individual transparencies. While labelling may be generated by the computer system, it is often more flexible to use conventional lettering techniques applied directly to the transparency.

The 35mm reversal microfilm image can be directly mounted in a 2 x 2 inch slide mount for use with conventional slide projectors. Color acetate filters are easily mounted with the microfilm to enhance the high-contrast image. The 35mm microfilm can also be used as a negative for optical photographic printing processes.

Several high contrast 8 x 10 inch prints from the calculus series are included in the Appendix.

### Summary

Undergraduate mathematics education suffers from a lack of descriptive visuals for supplemental use in the classroom. High precision computer graphics have been presented as a viable approach to illustrating abstract concepts and problems. The motivation and methodology for the calculus series of computer-generated illustrations have been discussed.

The calculus series has proven to be a successful prototype for computer visualization in the classroom. We feel that these techniques may be extended to several other educational disciplines for modeling abstract phenomena.

### Note

1. Correspondence relating to the content of this paper should be directed to Professor Warner.

### References

1. Ryder, G. G., "The PFORT Verifier," *Software Practice and Experience*, Volume 4, 1974, pp. 359-377.
2. Warner, J. R., "Design Applications of the MIDAS Graphics System," *Computers and Graphics*, Volume 2, No. 1, 1976, pp. 15-22.
3. Warner, J. R., "MIDAS—A Compositional Modeling System," Technical paper, Colorado University Computing Center, Boulder, Colorado, 1977.

## Appendix

### The Calculus Series Illustrations

The calculus series consists of forty-three illustrations germane to topics in introductory calculus texts. The actual illustrations have been produced as large plots, glossy prints, 35mm slides (both positive and reversal high-contrast images), and multi-color overhead transparencies. Each illustration models a difficult concept or problem common to most texts.

The selected illustrations described here have been presented to mathematics faculty at several universities and received with considerable enthusiasm. The full calculus series set has been successfully used in the single semester introductory calculus course taught at Colorado University.

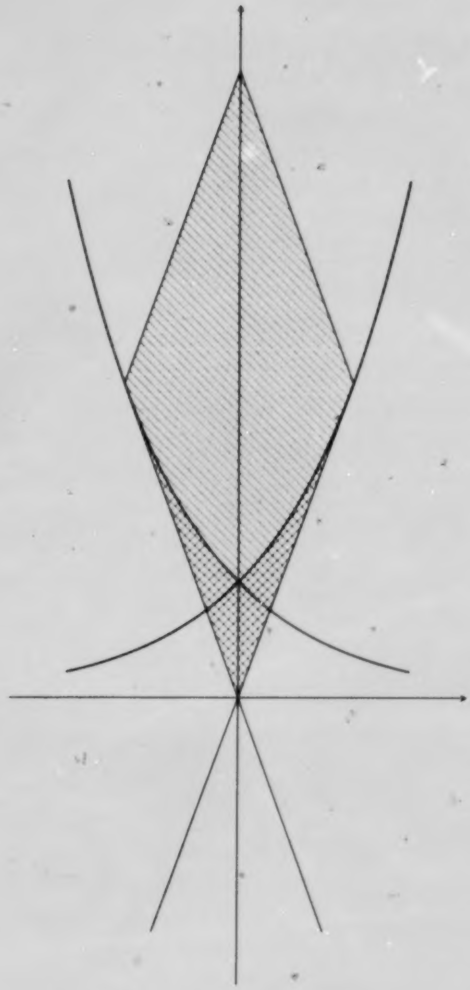


*Figure A-1*

This illustration typifies two-dimensional area problems involving the intersection of two areas. In using multi-color transparencies, the area of the rhombus and the area below the curves could be represented in different colors. The intersecting area becomes a third color when the transparencies are overlaid.

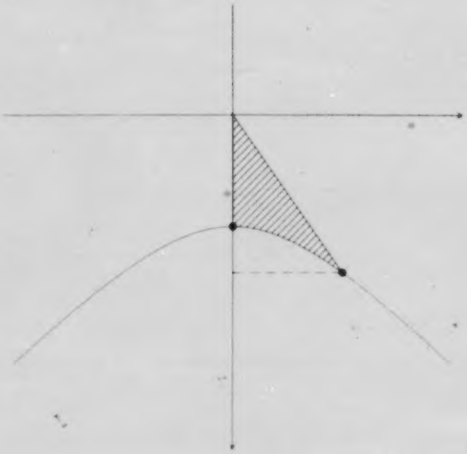


*Figure A-2*

Although contained in most calculus texts, this concept is rarely illustrated with exact precision. A parametric computer model permits precise representation and easy modifications to highlight different values on the curve.

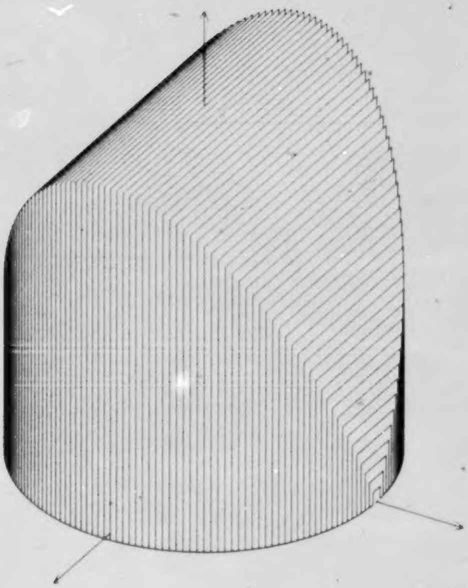


*Figure A-3*

A solid has a circular base. Every plane passing through the solid is a square (e.g., each "cross-section" is a square). Theoretically, the solid is composed of an infinite number of cross-sections. The computer model, with 200 cross-sections, illustrates both the three-dimensional shape and the approximation technique used in finding the solid's volume.

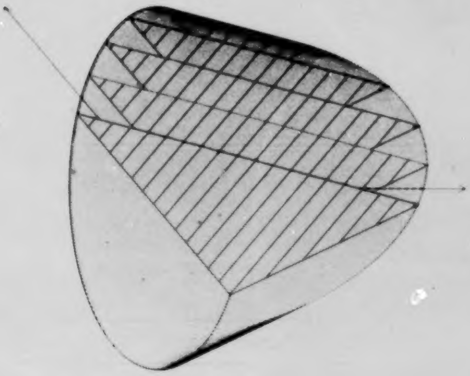


*Figure A-4*

This illustration is conceptually similar to Figure A-3. A solid is built from equilateral triangles on a circular base. The computer model is developed as two images; the solid, simulated by 100 cross-sections and five bilateral cross-sections. The classroom visual has either variable intensities or different colors to delineate the two components.

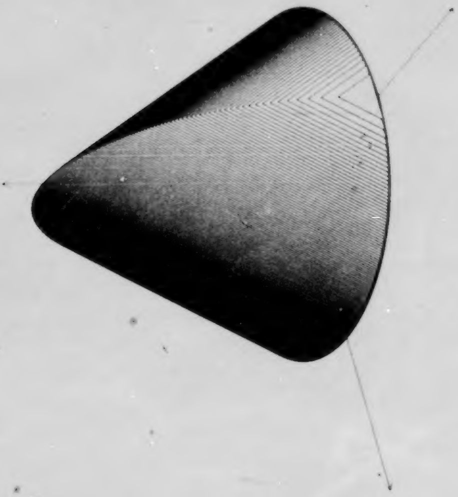


*Figure A-5*

Figure A-4 is viewed from another perspective. Once the initial simulation is complete, alternative focal points and/or viewing angles are trivial to implement.

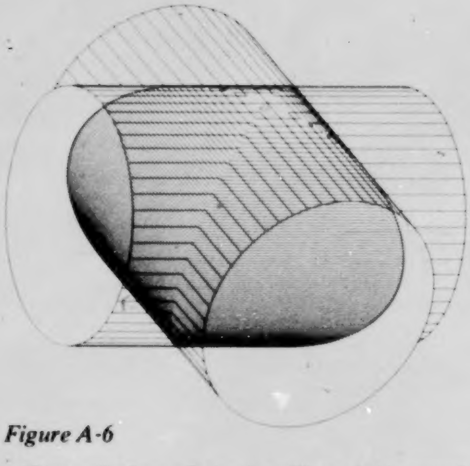


*Figure A-6*

Two cylinders of equal radii are intersected at right angles. The intersection (common area) of the two cylinders is exhibited. The solid formed by this intersection is easily described and visualized by its square cross-sections. In multi-color illustrations, the cylinders and intersections are in different colors. Each may be described separately, then combined to visually confirm the mathematics.

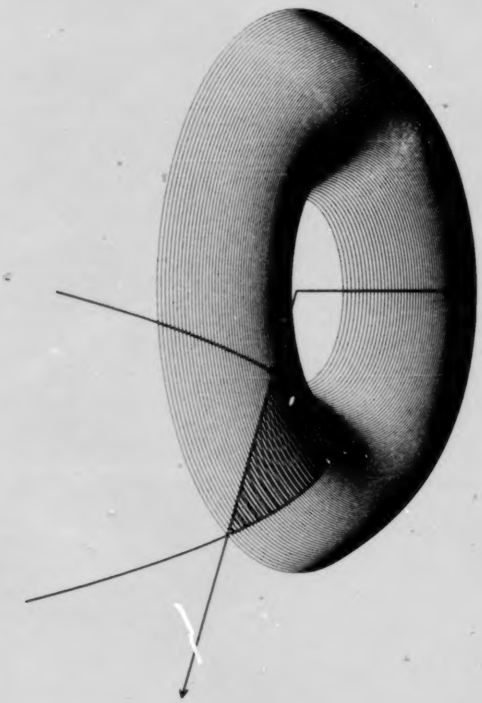


*Figure A-7*

A solid is generated by revolving the area between a curve and a line about some axis. The computer simulation overlays a single cross-section over the revolved solid. The solid is generated using circular disks, the same way a student must visualize the solid.

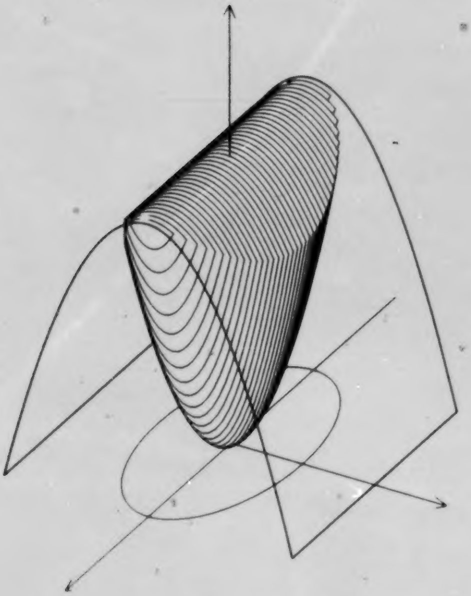


*Figure A-8*

Two quadratic surfaces (an elliptic paraboloid and a parabolic cylinder) are the boundaries of a solid. Evenly spaced cross-sections show both the solid's shape and the technique for finding the volume.

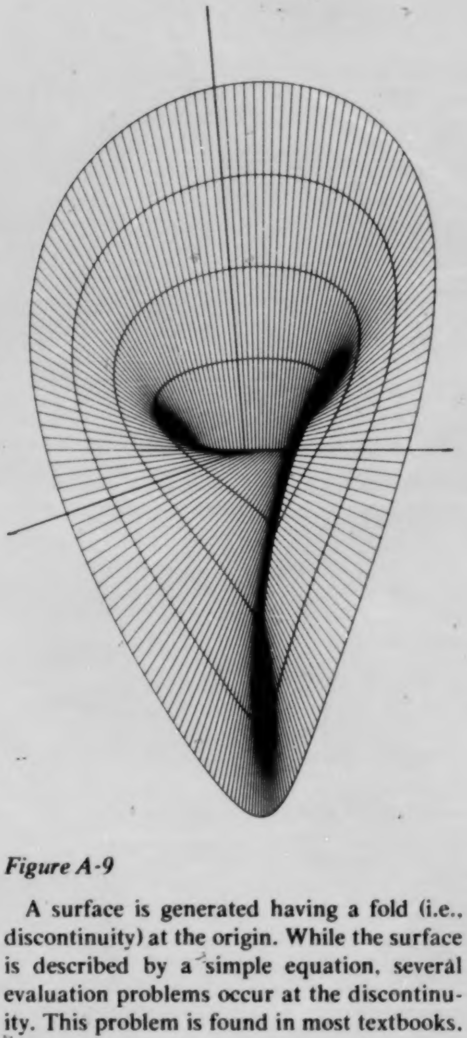


*Figure A-9*

A surface is generated having a fold (i.e., discontinuity) at the origin. While the surface is described by a simple equation, several evaluation problems occur at the discontinuity. This problem is found in most textbooks, but has never before been illustrated.

# Computer Art — A New Course Offering for General Education

**Grace C. Hertlein**
*Department of Computer Science*
*California State University, Chico*
*Chico, California 95929*
*(916) 895-6443*

## Background

During the past seven years, California State University, Chico, has offered a variety of beginning and intermediate courses in computer art for programmers, non-programmers, and for artists. This course derives its substance from the experience of working with hundreds of students from many disciplines, from many class levels.

An early paper, *The Artist and Computer Art*, from CCUC/3, 1972,[1] described an approach for artists. That material (and the other courses in computer art) has been drastically revised. However, in working with so many students, and obtaining good feedback from them, specific modules of instruction have been identified as useful, sufficiently good to retain, and these have been incorporated into the present course.

It is hoped that this definitive syllabus, with week-by-week lecture/discussion topics, together with laboratory experiences and assignments, will be a useful body of material for people who desire to implement a humanities-oriented computer course. Suggestions and ideas from the readers are welcomed by the author, and this material will become part of future planning for Computer Art for General Education.

## Attitudinal Changes Regarding Papers and Articles

During the past years, the author has engaged increasingly in writing and editing. Certain personal attitudinal changes regarding papers and articles have been noted as a strong personal change: going away from self-expression to consideration and communication of "useful" material for the reader. This means going beyond enticing "generalities" and beyond lists of materials used, to impart more material in one paper. This is particularly useful for many people, in that it eliminates a great deal of correspondence, trying to get more information from an author, stemming from an original paper. However, this method does depart from the traditional conference papers which appear to "lightly touch" on a subject, rather than developing the material in depth.

## Development of the General Education Course

The present syllabus arose from experimentation with other computer art courses, but more, from communication with the curriculum committee of the Department of Computer Science. The need became evident, to alter and greatly revise the upper division computer art course for computer science majors and minors, and to make that offering a technically oriented, practical offering. A description of this course, and a week-by-week syllabus is given in a paper submitted to SIGGRAPH 77, titled *Computer Art for Computer People—A Syllabus.*

Since the latter course and the present offering both are exposure courses for different kinds of people, there are related materials in both syllabi.

This class-tested offering is aimed at the freshman/sophomore student for general education credit. However, it is often taken by juniors, seniors, and even graduate students, people from many disciplines other than *computer science.* It may be taken by non-computer science majors and minors for lower division elective credit on a plus/minus (credit/no credit) basis.

### Rationale for the General Education/Art Development

A discerning reader will discover elements of the CCUC/6 paper on *The Microfilm Plotter and Computer Art,* from 1975.[2] There are also elements of a "Computers and Society" course, and from an "Introduction to Computers." Reference is made to the CCUC/5 paper, *Computers, Society, and Man.*[3]

What is the rationale for this combination of what is literally an "introduction" to computers, using computer art as the vehicle for instruction? Here are some answers:

1. Many institutions have courses titled "Computers and Society" and "Introduction to Computers." Too often the introduction courses evidence a traditional machine orientation, and allow limited computer use. The latter courses do not fulfill the liberal, interdisciplinary requirements for general education people. For technical people, these courses are superior, and are a prerequisite to further computer science offerings. The society courses have limited computer experience opportunities, because they are reading/discussion courses. Both these course offerings have valuable contributions to make in the undergraduate curriculum. More organizations (including ACM) are openly recommending a computer exposure course for every student, to prepare that person for living in a computer age.[4] This syllabus provides for such a generous computer exposure.

2. People who desire to use the computer for interdisciplinary, humanistic purposes require a special course tailored to their more general interests. The standard FORTRAN courses are too difficult for most interdisciplinary students. Further, they are not interested in a computer printout as a final product of their study. Additionally, these courses do not provide for the degree of personal expression, and the refinement of this expression, during a one-semester course. This tested syllabus affords personal expression and within its structure, has innumerable opportunities for personal decision-making and choices. It enhances decision-making capacities.

3. Graphics courses have a broader appeal for more kinds of people, from many levels. In a recent paper, Kai Chu remarks of results from traditional approaches, ". . . I found that they were ***not listening,***"[5] and contrasted the marked improvement in motivation, learning, and accomplishments when graphics were introduced.

The writings of Al Bork[6] repeatedly reveal the "power of graphics" to enhance learning and understanding.

In summary, a graphics-oriented computer course is a more powerful tool for learning. We are a picture-oriented, visual species, and we have been for thousands of years. Perhaps the strongest contribution of such a course is that people learn something about the computer by "doing." An experience course offers far more opportunities to understand and know, from John Dewey's idea of "learning by doing." Reading courses do not afford such experiences.

### The Approach

The approach used in this general education course is markedly different from the technically-oriented offering in which everything is presented in a technical manner. Although there is much analysis, problem-solving, and decision-making in this course, the method of presentation is informal, natural. Here are a few approaches tested in this syllabus:

1. ***Thinking About Computers in English.*** People in any given discipline tend to "speak in tongues," not understandable by the uninitiated. "Computereese" is a form of precise communication that regular people find difficult to understand. Faculty from computer science do not communicate in

English either, as evidenced by the comments of students. Gradually, the concepts of trying to think and talk about computers in English arose. In this course, students try to think about programming in English before programming, to understand in plain English, what is being required of the computer. It is easier to teach people to be more precise in English (or their natural tongue) than it is to teach them programming. To become more precise in speech and writing enhances communication between people, allows one to problem solve (and program) more readily.

2. *Canned Programs Used, Along with a Graphics Subroutine Library.* Students learn to use programs and plan personal variations, but they do not write totally new computer programs. Emphasis is on use, understanding. At first, canned duplicated decks are utilized, so that keypunching is at a minumum. After a few weeks, students are able to keypunch fairly well, and they can make changes easily. They are able to SYSCOPY their own decks and materials independently, and the transition is not difficult. Canned routines from the graphics library are used, and students merely pass parameters to attain personal variations. Details of this system are described in the week-by-week syllabus.

3. *Debugging Techniques Depart from Traditional Programming Classes.* Since canned routines are used, errors can be found more easily. Students are encouraged not to spend an excessive amount of time ferreting out errors, and after a reasonable amount of work in debugging, they may ask for special help. They are, however, encouraged and required to aim at independence. We have experimented in debugging via telephone, when the instructor is off campus, and writing at home. This works very well. Should this system fail, students will ride their bicycles to the instructor's home for a quick debugging session. The group learns far more than succeeding, going ahead, than from traditional debug methods.

4. *A Varied Menu Approach is Presented in Each Laboratory Session.* More is learned from presentation of varied materials within a single laboratory session. This is particularly true when the lecture hour is followed immediately by a two-hour laboratory period, making a three-hour class. During the last year, as an experiment, one group was assigned a lecture period followed by a two-hour lab, followed by a two-hour break, and a second two-hour lab, making a total of five instructional hours with the same students in one day. This was demanding for both students and the instructor, but it was highly successful, due to the intensity of the instruction, and the students found it stimulating, because of the variety of materials presented in one day. If participants followed the one day of instruction by obtaining special help during office hours (or via the phone), it worked remarkably well.

In an interdisciplinary class, presentation of varied materials is necessary. Further, such presentation avoids boredom.

5. *A Very Structured Approach is Used.* Coming from the humanities, the author has had a tendency to present materials in an open-ended flexible manner. However, experience repeatedly revealed that the fast students did very well, but those not in this category neither learned nor produced as well. Repeated feedback from students has urged a tight structure, with specific materials and assignments, and a faster pace. Whether we like (to admit it) or not, in many undergraduate institutions at present, students respond far more when pushed at least moderately. The progressive Montessori-like methods do not work at this level, for people will work harder for the firm, structured instructor. This was a great disappointment for the author to recognize this. However, a teacher's job is to "take people where they are," and then to take them beyond that point. The structured approach that provides for areas of freedom, does provide for growth and for self-expression. Students experience self-discipline.

6. *Parallel Processing of Information is Required.* This poses some difficulties for many students, for they are not accustomed to receiving, in one class period, input from diverse disciplines in one package. Perhaps in

the near future, when interdisciplinary studies have become a reality rather than an oddity, presentations of related (but dissimilar) materials will be more readily assimilated, because of student adaptation. The skill of considering several ideas simultaneously, and working this material into a synthesis is a practical, valuable development, regardless of discipline.

7. ***A Group Approach of Learning, Working, and Helping is Stressed.*** Education of the future will depart drastically from present approaches in use today. Content will not be the only change, nor reliance upon technological systems. Interdependent with content and use of technological systems is a need to develop human-relations skills that are satisfying and viable. The people in a class are at least as important as the subject, and the relationships between students in a class have been sorely neglected in many disciplines. There is a need to supplant individual competition with cooperation and communication on a person to person basis, and this can be translated as an educational experience on a student-to-student basis. With this frame of reference, students talk to each other, give each other suggestions. Using the group approach, they encourage each other, and literally teach each other. Carl Rogers' "unconditional positive regard" is the setting for this environment of person to person learning. ***There is far more learned by every person in such a class.***

## Summary

A review of the definitive week-by-week syllabus should be helpful to faculty who desire to begin a lower-division computer-oriented humanities course. There are advantages in using a tested syllabus, particularly using materials tested over a period of time. By definition of an interrelated structure for the entire semester, more is accomplished and gained, because of the alternatives that have been sifted, and the continuity of the retained material. One activity or topic leads to another related exploration.

It would be naive to assume that another institution would adopt the material in total. The tested material provides sufficient information for individual faculty adaptation, incorporating personal materials, and is presented in the hope that it will be useful to others.

## Week I

*Lecture:*

Slides of computer art are shown. Emphasis is placed on enjoyment of the many varieties of personal expression. National, international, student works are shown and informally reviewed by the instructor. The newest trends and ideas are very briefly related. Concept: The computer is an aid in creation, a colleague, not a protagonist. People of many types are able to use the computer to create personal art, and many of them would not be able to achieve the level of their art without a computer. Computer art is a new form of art that integrates science and art.

*Laboratory:*

1. Review of the syllabus, course requirements, etc., are thoroughly discussed by the group and the instructor. Identifying the tone of the class, as a group, with cooperation and open questioning replacing competition. Becoming acquainted with the group, working individually and as a group is emphasized.

2. Introduction to designing, using a continuous line design, or a design made up of two continuous line designs. Brief review of types of continuous line designs using slides, and their development by other artists and students. Brief introduction to design derivation techniques, to achieve half a dozen alternatives before listing the coordinates for the first, quick design.[8]

3. Introduction to cartesian coordinate system and pen values. Listing in an array the latter values. Checking each student to see that he understands the system. NOTE: In order to accomplish this, the design should not be in great detail—it is merely a step to begin using the system, to begin working with the computer.

4. Introduction to job control language in very simple English terminology, explanation of ways to punch cards in fixed format on data cards. Use of duplicated programs, so students will have very little keypunching to do in these first few weeks. Students type coordinates and pen values and input programs into the computer.

*Assignments:*

1. Study the inventory sheet of computer reactions handed out at the end of the last lab session, and think over personal ideas about the computer. Bring to next class session for discussion and review.

2. Collect pictures of art works from varied sources that have a strong personal appeal. Use photographs, Xerox copies, magazine and newspaper illustrations, advertisements, etc. Use in the second laboratory period.

3. Proof the first design on the computer semantically, then write a magnetic tape, and bring to class for the first lab session. Demonstration of the graphics system will allow students to proof designs on the cathode ray tube and on two sequential plotters.

4. Recall types of computer art that had a strong personal appeal. Art checksheets will carry this analysis further next week.

## Week II

*Lecture:*

Informal history of the varied computer arts: art graphics; designs for textiles; automated-executed weaving; choreography; computer-generated compositions and computer-executed music; computer-designed sculpture and paintings, etc. The prehistory of computer art is reviewed—mathematics and science in art, particularly in the early 1900's, and in the Bauhaus. Examples of the latter as illustrations of the art/science movement as a natural prelude to computer art. Changes in the types of participants, and the newest tendencies in computer art.[9] The future of the computer arts.

*Laboratory:*

1. Art checksheets, and a very brief perusal of the types of art, their sources, ways of programming and development. Using the checksheet to chart first reactions to this new medium. Date the checksheet, so sequential notes (in a second color) can be noted, as preferences change from exposure and experiences. Note reasons for preferences—beginning experiences in verbalization. List reasons on the back of the forms.

2. Review of inventory sheet of computer reactions: (a) What is a computer? (b) How does it work? (c) What can it do? (d) What can it not do? (e) The future of computers and man? (f) Processing vs. thinking on the part of man and computers? NOTE: This inventory sheet is deliberately used before seeing a film on computers, and the same sheet will be used again during the semester several times. Introduction to subjective, objective thinking processes, analysis. Discussion of reasons for reactions, and study of what types of thinking are involved, i.e., are ideas about computers based on non-exposure, fear, movies, etc.?

3. Film, "The Digital Computer" by the U.S. Navy. Review of the film, explaining in English computer terms, along with developing a computer vocabulary. Discussion of how a computer works, and its logical processes, along with its speed. Computer errors, and examples of their occurrence, reasons for, etc. Computer art is an artistic problem, in which decision-making occurs. Making out a class schedule is another form of problem, and decision-making takes place. Becoming conscious of problems, decision-making, solutions.

4. Introduction to a new subroutine, Factorit, a decrementing routine on the computer graphics library. Contrasting a subroutine with a program. Reviewing the subroutine, to note differences. Class brainstorming about innumerable ways to vary the first module, noting these in pencil on graph paper, then choosing from alternatives. Noting that this is another example of decision-making. The need to search for alternatives before making decisions is em-

phasized. Using the new subroutine in a new program.

*Assignments:*

1. Rewrite the inventory sheet of computer reactions in a composition.
2. Proof and run the second program on the plotter or CRT.
3. Assemble a three-ring binder with tabs for varied notes and ideas.

### Week III

*Lecture:*

Design derivation from varied sources: the environment, nature, man's mind, science, art, etc. Meaning of derivation, from a source. The need to recognize that ideas are input from sources and processed by man. Illustrations of student sketches, derivation exercises, along with work of Mondrian are presented via black and white slides. Demonstrating very briefly the system used (see 8), with experience this week by using the exercise system in lecture and in lab.

*Laboratory:*

1. Library research, in which students study graphics, and/or subject material of their choice, based on the illustrations brought to class. Exploration of opposites is briefly introduced: identifying what is not liked, what is pleasing, and noting degree of unattractiveness. Recognition that forms and patterns, sources are subjective preferences, and that no one form is better than another, although in art, there are fashions of forms and development. Instructor assistance in Library.
2. Design derivation exercises, executing the technique described above, in the NCC '77 paper. Addition and subtraction to input forms is accomplished. Learning to create quickly, spontaneously as a game. Learning to create without fears or inhibitions—just to do it, naturally. Very brief questions on creativity.
3. Introduction to subroutine Restore, a routine that allows the manipulated component (that was decremented towards a point) to be restored (original values called again), and decrementation towards a second or third origin may be achieved. Again, the game-like attitude, brainstorming about alternatives. Using a canned, duplicated program, keypunch up the very few cards needed to use this program. List the program for errors, and input into the computer. Proof graphically outside of class, for next week. (All routines are from the FORTRAN IV Art Manual.)[10]
4. Verbal sharing of rewritten ideas about computers. Note differences of opinions. Divide class into two groups. Briefly, spontaneously debate on pros and cons of computers from a list of topics quickly made up by students relating to the assignment. Then switch sides, giving opposite opinions. See how many ideas, both pro and con can be logically and/or subjectively mustered for debate.

*Assignments:*

1. Correct Program II, using Factorit or Restore, so that it is correct and attractive, to begin experimenting with art papers during Week IV. Begin to think of personal choices in colors.
2. Conclude design derivation exercises, so a new design (II) will be programmed and developed in class in laboratory session.
3. Mark design derivation sketches with preferences, and note in English, reasons for preferences. Note opposites, disliked sketches, analyzing reasons for such reactions.

### Week IV

*Lecture:*

Introduction to color, art materials, using student works, and works by other artists (actual graphics, textiles, etc.). Effects of color, color values, variant pen thicknesses, etc. Review of the color wheel, primary, secondary, tertiary colors, etc. Effects of analogous colors, and complementary colors, and how they may be used. Handout on material development. contemporary. traditional effects. etc. and how they are attained. Demonstration of color exercises is given.

*Laboratory:*

1. Demonstration of colors. using diverse colored papers. and how to attach them to the drum plotter. how to change pens. etc. Developing several finished graphics in color from student works. Analysing. when is a work finished vs. the exercise. and noting individual preferences regarding simplicity vs. complexity and gradations in-between.

2. Introduction to a random routine. Rfacput. that allows pseudorandomization of initial and last placement design component on the X and Y. along with minimum/maximum sizes desired. and number of modules to be placed on the paper. Looking at a few examples of work that contain randomizations.

3. Using planning sheets to achieve many alternatives by changing parameters before programming. Exploring opposites. large. small works; complex. simple works. etc.. attained by planning before programming. Using the new routine. again with a duplicated deck. and keypunching two or three cards to achieve a personal work. Inputting the program into the computer. Introduction to piggybacking programs. so that many may be run sequentially on the magnetic tapes.

4. Learning to think about programs in English is begun and continued throughout the semester. (People who work consistently with computers talk in "computereese." which the uninitiated cannot understand.) Expressing the more precise English. what goes on in a program. so that students can begin to write program statements in English to be able to debug their programs more readily. and to understand what they are instructing a computer to do. Development of computer vocabulary: fields. arrays. dimension statements. a program. a subroutine. do-loops. job control language. etc.

*Assignments:*

1. Using a brief demonstration in color exercises. students complete three pages of simple. identical drawings with twelve colored felt pens. Note color preferences. dislikes. and reasons for same.

2. Run a graphic on art paper before next week. and bring to class for review and sharing. Use preferred colors of paper and pens.

3. Debug new random program. and run on industrial paper (or art paper if it is pleasing).

4. Make notes in English on ditto sheet of new sample programs. and discuss during Week V.

### Week V

*Lecture:*

Creativity. inventivity. using in laboratory a film. and notes from Hertlein.[11]. Mueller.[12] Rothbart.[13] etc. Manipulation vs. creativity. along with a search for more precise definitions of creation. assemblage. execution. rearrangement. synthesis. etc. Deliberate brief exploration of ideas before seeing film. to note differences in film. Citing innumerable areas of life wherein one may be "creative. inventive."

*Laboratory:*

1. "Why Man Creates" is shown. This excellent film by Bass. in movements. in which specific ideas are explored. After showing of the film. explore specific ideas in each movement of the film. and discuss the essence of each portion. Relate creativity. inventivity in science to art. and contrast Einstein's statements. Edison's. Hemingway's. etc. Can creativity be assisted by the subconscious? Are creativity and inventivity "brute force" activities? Can the computer be creative. inventive?

2. Handouts. "Are You Thinking Machines?" by Arthur C. Clarke. and Turing's brief essay on the subject.[14] Brief overview of ideas in the two essays. Identifying what is meant by the term. "to think." Search for definitions. Comparisons of ideas between processing and "thinking." The role of the subconscious. synthesis. with excerpts from several sources. with discussion by the group on this area.

3. Introduction to Mirror, a graphics library subroutine. Use of the latter, to achieve new variations of previous programs using the decrementing and random routines. New programs are written, and keypunched (using canned decks), so that comparisons can be made with the first Factorit and Rfactput programs without Mirror, and those with reflected forms.

4. Open lab period for extra help, and fuller development of material presented and assigned.

*Assignments:*

1. Read thoroughly the two articles briefly reviewed in class. Write a three page paper on the topic. Review the paper, and edit it, labeling subjective, objective areas, and rearrange the material. Look for underdeveloped areas, and overdeveloped material. Prune if necessary. Hand both papers in for discussion and review during Week VI. Integrate materials, ideas presented during Week V.

2. Program Design II with Factorit, Mirror and Restore. Then do another program (using canned decks) for Rfactput, Mirror and Restore. Proof on industrial paper before next laboratory, and if artistic, run on art papers with colored felt pens.

3. Maintain notebook, adding new ideas from outside readings, class material, and new insights gained from discussions.

4. Choose a related, third design from sketches, to program during Week VII.

## Week VI

*Lecture:*

Computers in the humanities, their origins, present state of the art, with brief examples of music, slides of sculpture, illustrations from art exhibition catalogs, actual works, printouts of choreography, etc. The "humanities" in some instances include art, anthropology, archaeology, English, history, music, philosophy, religious studies, speech, drama, dance, etc. (There are differences in various institutions.) Review the handout, a listing of papers on computers and the humanities from a recent international conference.

*Laboratory:*

1. Discussion, integration of above ideas with student writing on thinking machines, the role of man; etc. Identify main ideas occurring repeatedly in student reactions and compositions, and list these. Students are to repeatedly note down and maintain a file of specific, personal reactions to discussions, films, etc. Excerpts from Sheldon Klein, automatic novel writing,[15] and Russian Folk Fairy Tales,[16] from recent international conferences.

2. The exercise vs. the finished graphic—studying works executed thus far. Responding as a group (and individually) to graphics, and verbalizing regarding color choices, positive and negative areas, dominant and subordinate elements of composition, etc. Noting on art checksheets in a second color, any changes in preferences and working habits at the present. When is a work finished? What are some principles of art that are used in computer art? Is art purely subjective: "I don't know much about art, but I know what I like" idea is banished. Trying to become more verbal, more precise in statements about art, and in other areas, and in communicating more.

3. Finding a personal direction through analysis of all work accomplished thus far. What are finished works? What graphics express personal ideas that are pleasing? What are the reasons for pleasure or subjective displeasure? Planning to revise all unsatisfactory work, with special help to program changes, revisions. Use of planning sheets to assist in revisions, and learning independently to revise programs, going beyond instructor's help.

4. Open lab period to reprogram needed changes resulting from above discussions and planning.

*Assignments:*

1. Make a portfolio to contain all flat art work. Have brief notes regarding preferences and development of all material.

2. Revise all graphics programmatically, so that finished art paper graphics result from revisions. All work should reflect a personal

direction (although that direction may be freely explored and changed during the semester).

3. Have all work up to date, to begin new programs next week, with a new design.

### Week VII

*Lecture:*

Interdisciplinary graphics, how graphics are used in other disciplines. Slides, illustrations, overhead transparencies are shown, with a presentation of graphics applications for innumerable purposes: oceanography, computer-aided design of automobiles, airplanes, ships, architecture, city planning, engineering, etc. A very brief overview of the systems that afford interdisciplinary graphics.

*Laboratory:*

1. Discussion, how are computers (and graphics) used in my major discipline? What is the potential of the computer in the student's career objective? How are computers and graphics being utilized at the present time in these areas? Motivation is given for research in this area, and a minor research paper to be undertaken within the next few weeks.

2. Program Design III, and duplicate personally any programs to be used via SYSCOPY, so that minimal keypunch occurs. Use of planning sheets to attain new variations. Since students are acquainted with a decrementing routine, a restoration of values subroutine, a mirror, and a random routine, they possess the ability to quickly program other designs, referencing known subroutines. New work should deviate from previous work, in that it allows a broadening direction. This is attained by planning sheets.

3. Library research on computer/graphics applications, with individual help from Reference Library Readings, and student desires for topical material. Suggestions from instructor, once student has identified the specific aspects of research in the major area of study. Finding periodicals and texts that are of suitable level during the reference period.

4. Open lab period to begin and complete planning and some programming of Design III.

*Assignments:*

1. Continue library research, and make note of references to be used. Make sure they are not too elementary nor advanced for mental stimulation.

2. Program Design III in at least two new ways, using different subroutines, and using other needed routines (as Mirror and Restore). The new graphics should depart from previous work, representing an exploration. For example, if the student has been working in a complex linear pattern, the new graphics may explore less complexity, and vice versa. Another example: if previous work is closed, without open spaces (void areas), new work should explore open areas of positive and negative spaces.

3. Graphically proof new work on industrial paper for Week VIII. Compare new works with previous graphics and note results, and preferences, along with reasons for success or lack of success.

### Week VIII—Midterm

*Lecture:*

Special showing of ruby laser graphics and computer music. As a break from midterm requirements, varied forms of computer-aided/executed music are played, with a laser set-up that affords red, moving graphics resulting from vibrations of the music. A brief review is given of different ways to use the computer in music, and also a description of how the laser graphics are achieved. Graphics are shown in a darkened room, with computer music audio.

*Laboratory:*

1. Guest speaker discusses lasers and their applications, and laser/computer uses in diverse disciplines. Question and answer period allows development of the subject.

2. Midterm self-evaluation form is given and completed, along with student input regarding revisions and changes for the course, with reasons for same.

3. Matting demonstration, with students learning to measure, cut mats with a beveled mat-cutter. Preparation of work for the portfolio, with special help.

4. Revisions of programs for graphics, so that work is more satisfying. Individual/group help in identifying needed changes. Completion of planning sheets to afford facile changing of programs.

*Assignments:*

1. All material accomplished thus far is due for review at the end of Week VIII. This includes (1) Notes; (2) Manual sketches and derivations; (3) Printouts of programs with machine sketches attached; (4) Art graphics, matted and unmatted; (5) Research on interdisciplinary applications, notes and references.

2. Complete midterm self-evaluation form and hand in with portfolio.

3. Take a second inventory sheet of computer reactions. Fill out the topical areas in detail. Note changes in attitudes resulting from experience. Write a one-page essay on growth and broadening of attitudes.

4. Have all material up to date, before gaining photography exposure.

### Week IX

*Lecture:*

Connotations of design, and graphic planning of augmented modules. A listing of combinations of development connotated by specific kinds of designs. This is an illustrated lecture, using students' tracings of original modules, accompanied by sorted slides. Example: a highly representational design generally uses varied mirror images, randomizations, serial imagery, rotations, etc. Lesser use of decrementing routines, offsets, radiations of pattern are shown, so as to retain the visual identity of the input pattern. Example two: An abstract pattern that allows for addition and subtraction of the input module has an unlimited number of possibilities, provided the top is different from the bottom, and the right from the left.

*Laboratory:*

1. Developing the lecture/demonstration, group work is first held in graphic planning of augmented modules. Before, all work was varied by a single design (module) that was manipulated in one way. At times the values were restored, and a second manipulation of the same routine took place. The augmented module is a module that is manipulated, and this enhanced, added-to, augmented module now becomes the design to be manipulated. The augmented modular approach affords many more opportunities for stimulating, professional-looking works.

2. Individual work in graphic planning is accomplished and begun in the laboratory session. (Again see NCC 1977 paper for details.) Students explore many alternatives, tracing the most satisfying augmented modules, and then note the ideas, coordinates, required to accomplish the new module. Example: a long-legged giraffe is mirrored on the X and the Y. The X coordinate is 2.0, and the Y coordinate is 4.2. The mirrored giraffes (four now instead of one) are randomized, serialized, etc. In addition, there are innumerable variations using just the mirror routine with the giraffe design.

3. Introduction to the subroutine augment that affords fairly simple use of the augmented design vs. the single design to achieve new works. Planning sheets introduce the idea, so that people have a chance to begin planning and programming in the lab session, so as to fully understand assignments and new ideas.

4. Open lab period for continuing planning and programming of the augmented module approach.

*Assignments:*

1. Continue planning outside laboratory sessions of graphic or visual planning to achieve new versions of Designs I and II. Explore in depth pleasing ideas, and sketch these new modules only. If a negative direction is noted, trace one example, and make brief written notes regarding this insight. (A negative reaction is often very helpful in finding positive directions.) Execute at least six

pages of new sketches, using all routines introduced thus far. Have coordinate, parameter notes required to make these changes.

2. Execute and proof one new program using the augmented module approach as a result of sketches and lab work.

### Week X

*Lecture:*

Photography explorations, including photographic manipulation of negatives, Kodaliths, use of a copy camera, Diazos, media combinations. A handout, a paper by the instructor, "The Microfilm Plotter and Computer Art" (see 2) forms the basis for this lecture, with additional ideas experienced since that time, and incorporated into the computer art classes. Actual works, and slides reveal processes used.

*Laboratory:*

1. Students survey work achieved and make decisions as to their favorite four or five best works to be photographed by the instructor in black and white. Half the works should be simpler modules, and the other half, complex developments of finished graphics. These works will be run in black and white for photography, on industrial paper.

2. Continued work with augmented modules, with review of planning, and sharing of preferred ideas. Analysis should be made whether the new ideas are a continuance of a personal art direction, or if they represent a departure, and to what degree, and how attained.

3. Programming of the augmented modules, with additional individual/group help in this more complex programming undertaking, so that students understand and can independently use this system in future works.

4. Writing assistance on interdisciplinary computer applications is given, based on notes accumulated thus far. Labeling paragraphs with subtitles, rearranging order of material, deleting, adding ideas that appear necessary are accomplished. Use of examples, rather than generalizations is given. Individual and group discussion of principles being used in this research paper. (Note: The course designed for computer science people requires an extensive, highly developed technical paper. This course requires less technical development, but more research into varied applications in the discipline, and it is not as lengthy.)

*Assignments:*

1. Complete and proof at least one more program using the augmented module approach. (Students now have two such programs.) Each program should be different from the others, and should not utilize the same subroutines.

2. Work on the research paper in depth. Draft one is due at the beginning of Week XIII for laboratory work.

3. If additional writing or programming help is needed, get that help from the instructor so that new work is understandable and executed with a minimum of difficulty. Do not belabor assignments and get "stuck," but rather receive help. If special appointments are required, make them with instructor.

4. Attempt spontaneous, creative approaches in execution of work, as presented during lecture/discussions to see if this approach is efficacious. Note difficulties and facile accomplishments.

### Week XI

*Lecture:*

Guest lecturer on computer poetry, uses of the computer in concordances, stylistic analysis of written material, etc. Readings from computer poetry, by the guest lecturer. Discussion period.

*Laboratory:*

1. Do a brief experiment in computer poetry, Haiku, assisted by the guest lecturer. Work in laboratory sessions with the instructor to complete the poetry assignment, using BASIC.

2. Open lab period to finish poetry work. Emphasis on personal directon, spontaneity, art as a form of play.

3. Introduction to varied offset routines, including offsets on the X/Y, and a routine that allows offset values to change, along with the scalar value (size) of the component design. Planning sheets facilitate execution of innumerable alternatives before programming. Use of tracing paper mode of visual/graphic planning allows visual perception of the precise offset values to use, and the number required for specific types of images. (See Laboratory session (2), individual work in graphic planning, discussed in Week IX, and in lecture for that week.)

4. Checking photographic slides by instructor of work on 35mm film, with students ironing, preparing their own slides for mounting and use in Week XII. Extra help on material presented so far.

*Assignments:*

1. Complete poetry programming, and debug for Week XII.
2. Complete planning sheets for offset routines, and program two very different works, departing from previous work accomplished during the semester. Proof and run on industrial paper.
3. Have slides and work ready for photographic session during Week XII.
4. Continue work on research paper.

### Week XII

*Lecture:*

Actual photographic demonstration of: Photographic manipulation of 35 mm positives and negatives, use of a copy camera, reversals, etc. Informal relating of the use of a copy camera in a special laboratory, with instructions on enlargement or decrement of images, along with combinations of materials and images, achieving reversals on photographic paper. Developing, making transparent sheet film (Kodalith) positives and negatives to use in Diazos, silkscreening, etc.

*Laboratory:*

1. and 2. Manipulation of negatives in a multi-station photo lab. Demonstration and actual experience in manipulating film images of computer art. Note: Half the images were simpler forms, and suitable for fairly active manipulation of multiple-exposure images (5-6); while the more complex negatives can be used in offsets, mirror images, etc. Students work with the negative slide, the positive slide, and the combination of the two in a bas relief form. Solar prints are also demonstrated and attempted. Concept: Computer-aided design can be taken back into varied forms of art, and these forms are particularly effective in photographic graphic arts.

3. Washing and drying prints made during laboratory hours one and two, with informal discussion by the group of specific effects attained by photographic manipulation, not available on cathode ray tubes and plotters. Reference to Herbert Franke's techniques[17] and other possibilities for further explorations.

4. Open mini lab period to complete any work unfinished from previous assignments. Individual and group help.

*Assignments:*

1. Catch up on any unfinished programming, debug, and run on industrial and/or art papers in finished form.
2. Proofread interdisciplinary graphics/computer applications research papers, and have ready to hand in and discuss during Week XIII.
3. Have Kodalith negatives ready for Diazo demonstration next week.

### Week XIII

*Lecture:*

Informal lecture/demonstration on serigraphy techniques, making a silkscreen, making a direct photo-silkscreen image using Ulano film, developing film, and actual printing of serigraphs, using computer-designed images. Brief history of serigraphy, its uses, types, with demonstrations on making, finishing screens, etc.

*Laboratory:*

1. In the same silkscreening lab. demonstrate the process making a quick edition of computer-designed prints; showing how a wall hanging is begun and completed (with samples of finished works); screening on plexiglas. and on two sides of Diazo images. etc. While screening. informal relating on types of film and inks (walter-soluble. oil-base systems). etc. Overprinting. underprinting are discussed and viewed as processes demonstrated in art material development (underpainting and overpainting) on the plotter with art papers and colored pens.

2. Return to the mini laboratory. and sharing of ideas from the interdisciplinary research papers. Students are required to make brief notes on cards. and these are permissible to use in a brief oral presentation of ideas and concepts gained from research. Papers are then handed in to instructor at the end of the laboratory sessions. and are used as back-up for questions by their peer group. Note: Too often research is accomplished by the student for himself and the instructor. Much greater learning is attained if varied materials are shared by the group.

3. Continuance of brief oral reports by students on interdisciplinary research.

4. If time permits. have open lab period. If not. use time to complete oral reports by the group. If overlaps occur. and several students have similar topics. have them gather briefly as a panel to summarize the material more broadly.

*Assignments:*

1. Complete all programming so that finished work is matted or ready to mat for final portfolio.

2. Complete all notes. and have binder ready for review. along with a section of computer printouts. machine sketches. etc.

3. Title all completed art works with personal names. On a card. list routines and materials used. Run an extra printout of the program. to hand in with the graphic. lightly tacked with rubber cement to the back.

## Week XIV

*Lecture:*

Untitled lecture (TBA) on new experiments by instructor. At this time. new work is being undertaken in automated choreography. working with dancers. and filming of the results for a conference. Another project is inputting varied forms of computer art into a synthesized television system and planning music for the resultant film. A history of the project. and descriptive details will accompany the lecture.

*Laboratory:*

1. Diazochrome demo in an engineering laboratory will show how to develop colored acetate Diazos using Kodaliths and acetate papers. including blueline and brownline papers. Combinations of colors. materials. etc. will be demonstrated. along with finished works.

2. Students work in blueprint room. making Diazos. and combining them with varied Diazo materials. photographic prints. art prints. etc. A supply of inexpensive blue. brownline papers. yellow and blue acetates are available at cost from the instructor. purchased wholesale.

3. Matting unfinished work. with group and individual instruction in finishing graphics. using regular. symmetrical mats or individually planned. irregular mats.

4. If matting is complete. go to open lab period. Generally. the last weeks of each semester are required for matting. plus last-minute help to individual students.

*Assignments:*

1. Finish matting. and check out a mat-cutter from the department office for overnight use. returning next day. so that others may share materials outside of class.

2. Finish running art graphics. checking for completeness of work.

3. Review notes taken during the semester and bring to next laboratory session for writing assignment. to be begun in class. on viewing the computer.

4. Get any last-minute help needed in any area introduced thus far.

## Week XV

*Lecture:*

Animated films highlight the end of the semester, with generally rented material by noted film-makers, along with brief materials available without cost from Bell Laboratories, etc. Sharing of films with others in the department cuts the cost of film rentals. A brief discussion of ideas presented in the film is given, along with philosophical considerations, ideas by the originator.

*Laboratory:*

1. Rerunning of films without sound often takes place, to analyze and discuss as a group, techniques of metamorphosis, dominant design changes, etc. (Sometimes films without the sound are run backwards, and are equally effective, accompanied by group discussion.)

2. Matting session, generally required by many in the group. Individual help in finalizing portfolios is given also.

3. Diazo work in engineering lab is generally requested also, to use with industrial paper graphics, photographic papers, etc.

4. Open book essay (beginning of final): viewing the computer, integrating materials introduced during the semester, personal research, attitudinal changes, etc. Students accomplish the listing of main ideas and brief paragraphs or sentences from notes and discussions. This is begun in class, so that the specific parameters of the assignment are understood (allowing for broad personal interpretations), so they may work on the essay before Week XII, finals.

*Assignment:*

1. Have portfolio and all work ready to hand in and share at finals time.

2. Work on essay assigned, and have this ready to hand in at the end of Week XVI.

3. Get any last-minute help needed, so that all work is handed in, and arrangements are made with instructor to photograph best works. Arrange for pick-up time of art works during Week XVI after finals, before leaving school.

## Week XVI

*Final Examination:*

Generally a two-hour block is reserved for finals. Several activities are emphasized:

1. Sharing of best works, displaying them, enjoying the works of others, discussing material, design development, checking the program on the back, etc.

2. Final self-evaluation form, a definitive self-grading form designed by instructor, along with feedback on the course, with suggestions for changes, together with reasons for change. Brief discussion with the group regarding needed changes in curriculum.

3. Portfolio examination, with a personal interview if desired with instructor to discuss progress, etc.

4. Arrangements for photography and pickup on an overnight basis. Sketches are photographed in black and white, and best finished works in color, by instructor for slide collection.

## References

1. Hertlein, G., "The Artist and Computer Art," *Proceedings*, CCUC/3, Georgia State University, Atlanta, June 1972.

2. __________, "The Microfilm Plotter and Computer Art," *Proceedings*, CCUC/6, Texas Christian University, Fort Worth, Texas, June 1975.

3. __________, "Computers, Society, and Man," *Proceedings*, CCUC/4, Washington State University, Pullman, Washington, June 1974.

4. Geisler, R. Gene, "Teaching Computer Science," *Computers and People*, July 1974, Vol. 23, No. 7, p. 2-3.

5. Chu, Kai, "Computer Graphics and High School Education," *Computer Graphics and Art*, August 1976, Vol. 1, No. 3, p. 18-22.

6. Bork, Al, "The Physics Computer Development Project," *Computer Graphics and Art*, August 1976, Vol. 1, No. 3, p. 10-17.

__________, "Learning through Graphics," *Computer Graphics and Art*, February 1976, Vol. 1, No. 1, p. 5-9, 30-31.

7. Rogers, Carl R. and Stevens, Barry, *Person to Person: The Problem of Being Human—A New Trend in Psychology*, Lafayette, California: Real People Press, 1967, 276 p.

8. Hertlein, G., "The Importance of Art Design Techniques and Material Development, Their Effects in Achieving Art Quality in Computer Art." Unpublished Paper Submitted to NCC 1977, Dallas, Texas, June 1977.

9. Hertlein, G., "NCC Art Review, Directions and Questions: The Role of the Computer in Computer Art." *Computers and People*, August 1976, Vol. 25, No. 8, p. 6-7.

10. __________, "FORTRAN IV Art Manual for Artists." Documents Division, Office of the Chancellor, California State University and Colleges, Long Beach, California, September 1976, 72 p.

11. __________, "Art/Science: The Union of the Rational and Irrational." *Proceedings*, T-5, June 1973, Zagreb, Yugoslavia.

12. Mueller, Robert E., *Inventivity*, New York: John Day, 1974.

13. Rothbart, Harold A., *Cybernetic Creativity*, New York: Robert Speller and Sons, 1972.

14. Dorf, Richard, *Man and Computers*, San Francisco, California: Boyd and Fraser, 1974.

15. Klein, Sheldon, (and others), "Automatic Novel Writing: A Status Report." ICCH/1, University of Minnesota, Minneapolis, June 1973, Technical Report #186, Computer Sciences Department, University of Wisconsin, Madison.

16. __________, "Russian Folk Tales." ICCH/2, University of Southern California, Los Angeles, April 1975.

17. Franke, Herbert, "On Producing Graded Black and White or Color Computer Graphics in Combination with a Photographic Technique." *Leonardo*, Vol. 7, 1974, p. 333-335.

# Computing in the Humanities: A Bicentennial Experiment

**Nell Dale**
*Department of Computer Sciences*
*University of Texas*
*Austin, Texas*
*(512) 471-7316*

The Department of Computer Sciences, University of Texas at Austin, began offering a course in computing in the humanities in 1969. For a long while the format remained basically the same. During the first half of the semester a nonnumeric computing language such as SNOBOL or BASIC was covered. The second half was spent surveying natural language processing applications such as concordances, content analysis and authorship studies.

Then in the spring of 1976, a different format was introduced in honor of our nation's bicentennial. The first assignment was for each student to choose two pieces of text, each approximately 150 words long. One was to have been written in 1976, the other was to have been composed in 1776. These passages were used throughout the semester. In addition to surveying language processing, the students used canned programs to do language processing.

During the second part of the semester SPSS was taught. For data, the students used statistics from their texts.

Outline of assignments:

*Choose text.* The sources varied widely: from *Newsweek* to *Redbook*, from *Poor Richards Almanac* to *Common Sense* by Thomas Paine. The only duplication was the Declaration of Independence.

*Examine graphic features of the text.* Each student was asked to make a list of all graphic features in his/her text. Each was then given a copy of a key-punch keyboard and told to devise a scheme to encode all the graphic features present. Since the students would be using a CDC6600-6400 System they had only sixty-four symbols to use. In addition the students were given reading assignments on the encoding of information.

In several days of heated class discussion, the following coding conventions were decided upon.

| *Graphic Feature* | *Key Punch Symbol* |
|---|---|
| period (.) | ¬ |
| question (?) | ∨ |
| exclamation (!) | ↓ |
| quotes (") | ≡ |
| quotes (') | ≠ |
| any accent mark | → following the letter |
| dash (—) | -- |
| apostrophe (') | * |
| new paragraph | new card, indent five spaces |
| italics | enclose in < > |
| bold face | enclose in ≥ ≥ |
| small type | enclose in ≤ ≤ |
| upper case | ^ following letter |

Additional conventions:

If a word ends in col. 72, leave col. 1 blank on next card.

If a word ends in col. 71, leave col. 72 blank.

Otherwise, continue from col. 72 to col. 1 of next card.

Identification information:

col. 73-74 blank
col. 75-77 students initials
col. 78 a 1 for 1776 sample
col. 78 a 2 for 1976 sample
col. 79-80 card number within sample

*Key punch text.* Each student had to punch his/her own text using the conventions decided on by the class. (One student, a classics major, asked if he could use old (800 BC) and new (100 AD) Greek texts. He devised his own encoding scheme and adjusted all the assignments to meet his case. However, the statistics on his texts were not included in the SPSS runs.) The first actual computer run was to list cards.

*CONTENT and KWIC.* Students each ran two programs using their texts. The first CONTENT produces an alphabetic word list with frequency counts, average word length, type/token ratio, and the number of times each special symbol occurs. KWIC produces a concordance with the card image as context. Reading assignments during this period included articles on concordances, authorship studies and stylistic analysis.

*Fill out Table.* Each student had to fill out the following table.

| *Fill in the following table* | 1776 | 1976 |
|---|---|---|
| Number of paragraphs | | |
| Number of words | | |
| Number of sentences | | |
| Average word length | | |
| No. of unique words | | |
| Number of prepositions | | |
| Number of coordinating conjunctions (and, or, but) | | |
| Number of 'a's | | |
| Number of 'the's | | |
| Number of uppercase letters | | |
| Number of personal pronouns—male | | |
| Number of personal pronouns—female | | |
| Number of personal pronouns—neuter | | |
| Number of infinitives | | |
| Number of 'there's | | |
| Number of color words | | |
| Number of commas | | |
| Number of semicolons | | |
| Number of colons | | |
| Number of dashes | | |
| Number of uses of verb 'to be' | | |

Some of the information was available directly from the print out sheet; some could be calculated directly, and some required interpreting the concordance. For example, the number of infinitives could only be calculated by checking each "to" in the concordance.

Several items in the table generated interesting class discussions and lead naturally into the problems of syntactic analysis and ambiguity. For example, what do you consider the pronoun I? Does it take the gender of the author? Can it be determined automatically? Just what are color words—just names of colors? Or would 'darkening' be considered one?

*Keypunch table.* The students were given a format description such as SPSS uses and told to keypunch their data. (By this time they were learning SPSS.) We took all of their cards and ran a simple frequency on each variable. The assumption was that if the range and average seemed reasonable, the keypunching was correct for that variable. For the variables that seemed suspicious (range in number of paragraphs: 1-736), the students checked a listing of the data to find the error and the guilty party had to correct his/her own. Some had to keypunch their cards three or four times and still made mistakes. This lead into a discussion of data format design and possible ways of making it easier to code and check.

When the frequencies all looked reasonable, the data was put on permanent file for the next stage of processing. There were thirty-four cases; seventeen from 1776 and seventeen from 1976.

*Statistical runs.* The students' SPSS project was to choose three variables and run a set of tasks to investigate each variable individually and then the relationships among the three variables.

While learning SPSS they were given a few lectures on basic statistics including levels of measurement, single variable statistics, crosstabulations, and comparison of sample means, again using the data collected from the two sets of text. This last assignment gave them a good workout using the basic procedures in SPSS. In addition, they had to demonstrate their grasp of basic statistics in their choice of measures and their interpretation of the results.

## Results

The actual textual experiment was far too uncontrolled to produce any valid results. However the number of sentences per 100 words, the number of infinitives per 100 words, and the average word length appear to be significantly different between the two samples.

More importantly was how the experiment worked as a teaching device. This was very successful. The students encountered and solved problems themselves instead of only reading about them. They generated a concordance and used it to solve a problem. They discovered for themselves how difficult it is to be precise even on such a simple task as counting the number of prepositions given a frequency list. They used the computer as a tool for some tasks and discussed the processes involved in computerizing some tasks they did by hand. They learned to use the machine, but not to program. However, they learned more about the power and limitations of the computer than those students who take a basic programming course.

The only complaint that was voiced was that no conclusive results could be seen from the experiment.

In closing, the author would like to reprint a poem which one of the students wrote.

### THE SPRING OF SEVENTY-SIX

by Sharon Hobbs

Sojourn with me through the humanist land
where computer software strikes up a band
and the bicentennial gala began
with the dishonor of a famous man.

The logic followed was perfectly clear—
computers are rational creatures (though queer).
Now listen my children and you shall hear
the coverup story of that infamous year.

First came hypothesis to delineate,
and appropo' measures to disambiguate.
Next came a function to discriminate,
telling which author to eliminate.

In bits and parcels the printout unfolds;
how Madison wrote as prior he had told,
How Hamilton lied—though his death was bold,
his encoded habits just don't fit the mold.

Now we move on to Part Two of our story:
SPSS and the death of the Torries.
Statistical Programming truly is gory,
the language alone would embarrass Old Glory.

Start with the File Name and Variable List,
then Input Medium (Cards, if you wish).
Subfile List follows if you do insist.
The setup continues—you get the gist.

You think you have finished—but now Recode.
If you use Frequencies—try General mode.
The computer will quickly relinquish its load.
Now *you* have the output to try and decode.

There's Chi-Square and T-Test for finding a sample.
And tables output by Crosstabs are ample.
Condescriptive will give you a handle.
Just ever remember—statistics' a gamble.

# Umlaut, ß, and Learning German Verb Conjugation by Computer

**L. Allen Viehmeyer**
*Youngstown State University*
*Youngstown, Ohio 44555*
*(216) 742-1851 Ext. 352*

As a valuable aid to learning German verb conjugation the expediency of computerized drills is undeniable. While this axiom may be generally acceptable, implementation is not always easy. Computer programming is rarely a part of the foreign language teacher's background.

We had heard about CAI aspects of foreign language instruction for several years, but had not considered attempting this approach to language teaching until our department decided to explore the possibilities of introducing mini-courses and competency-based courses into the curricula. Although we knew absolutely nothing about computers, we decided to learn how to write CAI programs so that the students could work independently on their problems. We went to our computer center director who pledged his full support and advised us to work with one of the student consultants. In discussing our plans, the student consultant informed us that we had no CAI languages at our disposal and that we should write our programs in PL/I.

Our facilities consist, in part, of an IBM vm/370 with IBM 2741 hard-copy, IBM 3277 display, and NCR 260 portable, hard-copy terminals. Of course, these terminals have keyboards designed for typing only English alphabetic characters. Since German has alphabetic characters not found on the standard American keyboard, modifications must be made so that these German characters can be used accurately. Correct spelling of the German word is a pedagogical concern. In the first programs we wrote the German umlated vowels ä, ö, ü and the special character ß were respelled as ae, oe, ue, sz, respectively. Although this respelling works nicely and is not bothersome to advanced students, it is pedagogically unsound for the beginner, who often has transfer problems. The student who learns to type ICH MUSZ, WIR MUESSEN at the terminal often has problems when he has to write ICH MUß, WIR MUSSEN in an exam.

This purely academic problem has been surmounted by the installation of a conversion kit on the IBM 2741, which allows the normal IBM selectric typing element to be used in place of the standard typing element for the 2741. Two 2741's were converted and German students are instructed to use those terminals. When a student wants to use the German languages drills, he simply replaces the standard English typing element with the German one. He has a small chart which informs him were ä, ö, ü, and ß are located on the keyboard. With our enrollment it would be unusual for more than two students to use the German programs at any one time. If a student happens to use one of the non-converted terminals, e.g., an IBM 3277, data sets with the alternate spellings, i.e., ae, oe, ue, sz, are loaded by an exec which monitors the type of terminal being used so that the student can work at that terminal.

Once the 2741 is converted any typing element manufactured for a selectric typewriter may be used on the terminal. Typing French, for example, is less awkward with the availability of ç, however, é, è, û, etc., can be typed only by backspacing—as is the case, of course, with the selectric typewriter. One irritating aspect of backspacing on a computer terminal is that there must be consistency on the part of the programmer and the user: e backspace does not match backspace e.

Having discussed the major hardware problems, the author would like to describe two interactive PL/I programs to assist the student in learning German verb conjugation. In writing these programs two main objectives were kept in mind: universality and versatility.

The first program, called CONJ, meets the requirements of unversality in that the data set consists of more than 350 verbs (with plans to add more) selected on the basis of frequency rather than appearance in a particular textbook. CONJ meets the requirement of versatility in that it allows the student to select the verb he wants to learn, and he can practice any one or all of the three persons singular and plural: ich, du, er (sie, es), wir, ihr, sie. He can type the forms in any order and can repeat them indefinitely in any order.

Once the student has accessed CONJ and read the instructions for the drill (note that this is not a lesson), he enters the infinitive form of the verb he would like to conjugate. If that infinitive is not found, the student is asked to check the spelling of the word he typed. If he did spell the word correctly, he is informed that that verb is not in the inventory, and that he will have to choose a different verb (with more than 350 different verbs students rarely come up with one that is not in the table). Once the student has keyed to a particular verb he may conjugate it any way he wishes as long as he types a subject pronoun, space, and corresponding finite form, e.g., ICH GEHE. If his conjugated form is correct, the computer responds with the message SEHR GUT! NUR WEITER, BITTE. If the form is incorrect, the computer returns X... IS INCORRECT. TRY AGAIN. If the student fails to write the correct form on his second attempt, the correct answer is displayed.

The student may move on to a different verb at any time as long as his last entry was correct. He never has to do a complete conjugation (textbooks vary considerably the order of introducing subject pronouns and irregular verbs). In order to go on to a different verb, the student must type NEW so that he can key to a different verb in the table. If the student forgets to go through that procedure and happens to type correctly a conjugated form of a verb other than the one he is keyed to, a check is made, and he is informed: X...IS CORRECT, BUT YOU SWITCHED VERBS. RETURN TO Y... OR TYPE NEW. This control of keying to a particular infinitive is necessary. Otherwise incorrect entries return a null string.

If the student cannot remember the verb endings, he can type HELP at any time and invoke a subroutine which displays the verb endings. This program is infinite and must be terminated by the student who can do so at any time by typing HALT.

The alternate program for drilling German verb conjugation, NOCH, has the same universality as CONJ (they use the same data sets), but it does not give the student quite the same freedom of choice—perhaps it is more challenging.

Once the student has accessed NOCH, a subroutine seeded by the clock randomly generates an infinitive for conjugation. The student has the option of conjugating that verb or asking for a new one. This option is important for beginning students who may not be familiar with the verb or its type of conjugation (reflexive, modal, etc.). If the student agrees to conjugate the verb, he is presented with a subject pronoun, for which he must supply the corresponding form, e.g., DU is displayed, and the student enters GEHST. If the student has not had that form (du and ihr forms are introduced quite late in some textbooks), he may pass and the next pronoun will be displayed. As usual, if the student does not supply the correct form on his second attempt, the correct form is displayed. Once the computer has asked the student to supply the correct form for all six persons, a new infinitive is randomly generated. This program, too, is infinite and must be terminated by the student.

Although NOCH is not conceived of primarily as a test, it does perhaps test the student's knowledge of conjugation rather than provide an opportunity for drill.

Both CONJ and NOCH make use of the four basic German verbal forms: present, simple past, general subjunctive, and special subjunctive. The student chooses the tense or mood he wishes to practice.

Student utilization of the seventeen drills written so far on various aspects of German grammar has not been as great as the author had hoped. We have a limited number of terminals and German students must vie for them with all the students on campus who are involved with various aspects of the computer facilities. Youngstown State University being an urban, commuter campus, students often have tight schedules and terminals are not always available when the students have time. Most German students are not familiar with computers and shy away because everything seems too complicated (students in scientific German are an exception). To make access to the German materials as easy as possible, we have written an exec, which, once the student has logged on and implemented CMS, formats the disk and loads the program and data set for the student. The student calls up the drill he wants merely by typing in its number.

Despite somewhat low utilization, we have been rewarded for our efforts on several occasions, however, when students have come back from their first encounter with the computer and have favorably and enthusiastically reported their experiences to their classmates.

# Using Networks to Net Results

**John P. Tucciarone**
*Mercy College*
*Dobbs Ferry, New York 10522*
*(914) 693-4500*

## Introduction

This paper is an outgrowth of a course in introductory computer programming given at Mercy College, Dobbs Ferry, New York, during the spring of 1976. There were two sections of the course, each taught by the author. The enrollment in both sections totaled fifty-one and consisted mainly of freshmen and sophomores whose ages ranged from eighteen or nineteen years on up to fifty-three.

The course was open to all students and it was devoted to instruction in the fundamentals of programming in BASIC, with students using remote teletype terminals to run programs. In addition to classroom lectures and "shop sessions" during which the author personally supervised the running of programs, the students were assigned a total of ten programs to complete in the course. These programs were selected to expose the students to the varied applications of BASIC. The programs included: looping exercises, character string manipulation, financial examples, matrix manipulation, and game simulation. Some of the discussions and assigned programs were selected directly from the text, ***Basic Programming***, by John G. Kemeny and Thomas E. Kurtz.

In addition to the seven programs whose topics were described above, the students were encouraged to select the remaining three from applications in other courses or from the remaining topics in the text, which had not been treated in class lectures. Some class time was devoted to the highlighting of several of these topics.

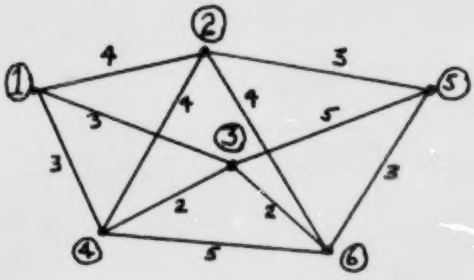
The topic which seemed to arouse the greatest interest was the Kemeny-Kurtz treatment of a critical path-network flow problem. The majority of the students in the course were planning to take business-oriented courses as part of their undergraduate curriculum, and approximately one-third were planning to major in business. Perhaps that fact explains why such interest was generated in a problem which would normally appear to a layman to be difficult to solve, albeit timely and practical. The author used the opportunity afforded by the curiosity of the students to discuss several elementary network problems. The author discovered that after a brief treatment of algorithms and suggestions for their implementation, and the organization of data, most of the students who elected to work on the network problems were able to write efficient programs. This paper is devoted to that experience.

In the following, three network problems are treated. The selection of these was made with two criteria in mind. First, that the problems be practical and easy for a student to identify with. Second, that simple algorithms existed for the solution of each, so that a reasonably elementary program could be written to model the algorithm. With these criteria in mind, the author selected: (1) A Minimum Tree in a Network, (2) A Minimal Time Path Through a Network, and (3) A Critical Path Through a Network.

## A Minimum Tree in a Network

*Problem:* Given a network of cities with interconnected highways, find the cheapest way to upgrade the existing roads so that each city is connected to every other by an upgraded route, and without the construction of new roads. That is, seek a tree without closed loops which includes all the intersections (cities), and minimizes the sum of the distances represented by each road (or edge) in the network.

Since the students were totally unfamiliar with the methods of linear programming, a simple approach was suggested for the solution of the problem. All the edges in the network were arranged in increasing length. The shortest was then chosen first and to that was added the shortest of the remaining edges which had one node (or intersection) in the tree, but not both. The solution tree is then complete when all nodes are included in the tree. The following example was considered:



**Network I**

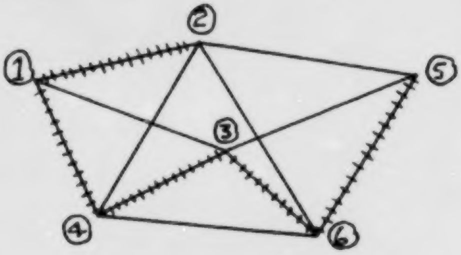
**The numbers in circles indicate cities and the remaining numbers indicate the distances between them.**

The edges were arranged in matrix form producing an 11 by 3 matrix in which the first and second columns represented the two nodes forming each edge and column 3 represented the length of that edge. The matrix is ordered on column 3, that is, the higher the row number in the matrix, the longer the edge. Otherwise, there is no priority as to which nodal end of an edge is listed in column 1 or in column 2.

Student written Program I (see Appendix) produced the following listing:

```
3 4 2
3 6 2
1 4 3
5 6 3
1 2 4
```

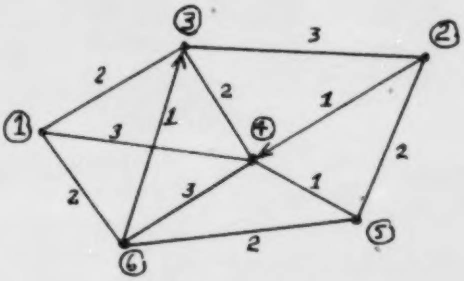
which is interpreted as meaning that the edges formed by the nodes in columns 1 and 2 form the minimal tree and the sum of the numbers in column 3 gives the total length of the tree, 14. The minimal tree is depicted as follows:



## A Minimum Time Path in a Network

*Problem:* Given a grid of streets and intersections in a certain city with the time needed to traverse a given street (from one intersection to another) at a given time of day, find the shortest time path from one intersection to any other.

The minimization could well involve distance rather than time but for students already familiar with the traffic congestion of metropolitan New York City, the minimum time path seemed to be the more appealing choice. The following example was considered:



**Network II**

**The numbers in circles represent intersections (randomly numbered), and the remaining numbers represent the time required to traverse that edge. All edges (or streets) are assumed to be two-way unless indicated one-way, such as 6-3, and 2-4.**

We start by arranging the data in two matrices, A, the intersections, and B, the edges. Matrix B, a 20 by 3 matrix, contains the 20 edges, counting two-way streets twice and one-way streets once. Column 1 contains the intersection number of each edge starting with 1, then 2, etc., in numerical sequence. Column 2 contains the time value for each, and column 3 contains 0 for each entry except when that entry marks the last edge having a given starting intersection number, in which case column 3 contains a 1. Thus, a search of column 3 in matrix B will yield groups of edges having the same starting intersection.

Matrix A, a 6 by 3 matrix, contains the 6 intersections. Column 1 contains the associated row number of matrix B which marks the start of the group of edges having a starting intersection equal to the row number of A. Column 2 contains the time values which are set at 0 for the intersection from which minimal time paths are sought. The remaining entries in Column 2 consist of time values which exceed the possible total for the entire network. In the example under consideration, 50 was selected as the starting value. Column 3 is set at the start with a value of 1, corresponding to the intersection from which the minimal time paths are sought, and 0's elsewhere.

For the example, matrices A and B are as follows:

*A*

| | | |
|---|---|---|
| 1 | 0 | 1 |
| 4 | 50 | 0 |
| 7 | 50 | 0 |
| 10 | 50 | 0 |
| 14 | 50 | 0 |
| 17 | 50 | 0 |

*B*

| | | |
|---|---|---|
| 3 | 2 | 0 |
| 6 | 2 | 0 |
| 4 | 3 | 1 |
| 3 | 3 | 0 |
| 4 | 1 | 0 |
| 5 | 2 | 1 |
| 1 | 2 | 0 |
| 4 | 2 | 0 |
| 2 | 3 | 1 |
| 5 | 1 | 0 |
| 6 | 3 | 0 |
| 3 | 2 | 0 |
| 1 | 3 | 1 |
| 4 | 1 | 0 |
| 2 | 2 | 0 |
| 6 | 2 | 1 |
| 1 | 2 | 0 |
| 3 | 1 | 0 |
| 4 | 3 | 0 |
| 5 | 2 | 1 |

The algorithm consists in starting at a given intersection and going from there to all those intersections which can be reached from it, calculating the time to traverse, and then to successive intersections. If an intersection is reached which has been reached before, a new travel time is substituted if it is smaller. When all the intersections have been processed, the algorithm terminates.

Student written Program II (see Appendix) produced the following listing:

```
0 0
5 0
2 0
3 0
4 0
2 0
```

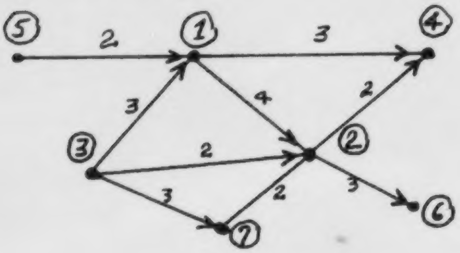
which indicates that the shortest time path from intersection 1 to those numbered 2, 3, 4, 5, and 6, is given by the values in column 1. That is, the shortest time path from intersection 1 to intersection 2 is a total of 5 time units.

To find the actual route once the minimal time is known, the following procedure was used. We work backward from the intersection one desires to reach by going to the pre-intersection whose time value differs from the time value up to the current intersection by an amount equal to the time value for that street. Of course, this procedure should be modified to include those cases where the actual minimal time path is not unique. In our example, the minimal time from intersection 1 to 5 is 4 units, but there are two minimal paths, namely, 1-6-5 and 1-4-5.

Student written Program III (see Appendix) produced the following listing when the intersection 2 was input: 2-3-1. This indicates that the minimal time path for a time of 5 units from intersecton 1 to intersection 2 is the path 1-3-2.

## A Critical Path Through a Network

*Problem:* Given a project with a fixed number of events to be completed and a process or activity which culminates in the completon of each event, several questions may be asked. In the directed network, directed since we assume there is some precedence in the flow through the network, what is the shortest time within which the entire project may be completed? What are the earliest times or latest times each event may be completed without affecting the best time for completion of the entire project? The following example was considered:

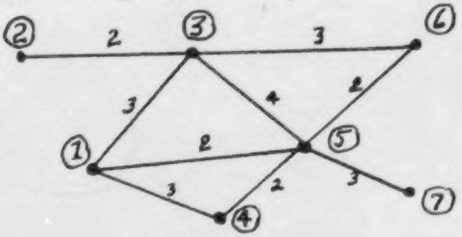


**Network III**

**The events are numbered in circles and the directed edges indicate the intervening activity with the time for completion of each marked.**

First we proceed to renumber the event circles so as to obtain a sequential topological ordering of the network. This process consists in the arranging of the data into two matrices, E and A. In the example, matrix E is a 7 by 2 matrix. Column 1 contains the row number of the activity matrix A in which the edges having a starting event number 1, 2, etc. begins. Column 2 contains the number of predecessors that each event has. Matrix A is a 9 by 2 matrix. Column 1 contains 0's everywhere except for 1's to indicate the final edge of a group whose starting event corresponds to that row number in E. Column 2 lists the successor event for the particular edge.

Student written Program IV (see Appendix) produced the following listing: 3-5-1-7-2-4-6, which indicates the correct topological ordering of the events in the network. A renumbering of those events produces:



We are now ready to find the earliest (latest) dates on which each event may be completed. We arrange the data in one 9 by 3 matrix, S. Columns 1 and 2 in S represent the nodal ends of each edge and column 1 is arranged according to the topological order determined above. Column 3 contains the lengths of each activity. The algorithm then proceeds by assigning a value of 0 to each of the events 1 and 2 since they have no predecessors in the network. We then process the list once in descending fashion, adding to the current value of an edge (given in column 3) the value of the second node (given in column 2). If the result is greater, we replace. After the list has been processed, we select as the value for each event or node, the largest value from column 3 which is associated with the occurrences of that node in column 2. In the example, matrix S has the form:

$$\begin{array}{ccc}
1 & 3 & 3 \\
1 & 5 & 2 \\
1 & 4 & 3 \\
2 & 3 & 2 \\
3 & 6 & 3 \\
3 & 5 & 4 \\
4 & 5 & 2 \\
5 & 6 & 2 \\
5 & 7 & 2
\end{array}$$

Student written Program V (see Appendix) produced the following listing:

```
EVENT 3 HAS VALUE 3
EVENT 4 HAS VALUE 3
EVENT 5 HAS VALUE 7
EVENT 6 HAS VALUE 9
EVENT 7 HAS VALUE 10
```

We interpret these values to mean that a total project time of 10 units are required and column 2 gives the earliest times when each event may be completed. A similar procedure may be used to calculate the latest times.

## Comments

The author found that a topic which is not normally treated in an introductory course in programming turned out to be one which elicited active interest. The course had no prerequisites and thus attracted students with a varied degree of quantitative competence. The author found, however, that most students were able to handle the programs effectively once the ideas were presented in a concrete framework and the algorithms were carefully explained.

The results raised two questions which prompted the author to write this paper. (1) Given the non-academic sophistication of many college students today, might a topic such as networks, usually considered advanced, be fruitfully studied at the introductory level? (2) For students who may be weak in arithmetic skills, might "on the job training" in the nature of actual problem-solving and program-writing improve those skills? Based on the experience of this course, the author is inclined to answer "yes" to each. Once the interest of a student is aroused and he or she feels the empirical need for a solution to a problem, a need which he or she can relate to, and understand, very little more in the way of motivation is needed from the instructor.

## References

Kahn, A. B. "Topological Sorting of Large Networks." CACM, 1962, Vol. 5, pp. 558-562.

Klein, M. "Scheduling Project Networks." CACM, 1967, Vol. 10, pp. 225-230.

Knuth, D. E. Fundamental Algorithms, Reading, Mass., Addison-Wesley, 1968.

Nicholson, T.A.J. "Finding the Shortest Route Between 2 Points in a Network." *Computer Journal*, 1966, Vol. 9, pp. 275-280.

## Appendix

*Program I*

```
010 DIM S (11.3), M(6), T(11)
020 MAT INPUT S
030 LET K=C=1
040 LET M(1)=S(K.1)
050 LET M(2)=S(K.2)
060 LET T(C)=K
070 LET L=2
080 LET K=K+1
090 IF K=12 THEN 290
100 FOR I=1 TO L
110 IF S(K.1) M(I) THEN 200
120 FOR J=1 TO L
130 IF S(K.2)=M(J) THEN 280
140 NEXT J
150 LET L=L+1
160 LET M(L)=S(K.2)
170 LET C=C+1
180 LET T(C)=K
190 GO TO 280
200 NEXT I
210 FOR I=1 TO L
220 IF S(K.2) M(I) THEN 270
230 LET L=L+1
240 LET M(L)=S(K.1)
250 LET C=C+1
260 LET T(C)=K
270 NEXT I
280 GO TO 80
290 FOR A=1 TO C
300 PRINT S(T(A).1):S(T(A).2):S(T(A).3)
310 NEXT A
320 END
```

*Program II*

```
010 DIM A(6.3). B(20.3)
020 MAT INPUT A. B
030 FOR K=1 TO 6
040 IF A(K.3) 1 THEN 150
050 LET X=A(K.2)
060 LET Y=A(K.1)
070 LET Z=X+B(Y.2)
080 LET D=B(Y.1)
090 IF Z A(D.2) THEN 120
100 LET A(D.2)=Z
110 LET A(D.3)=1
120 IF B(Y.3)=1 THEN 180
130 LET Y=Y+1
140 GO TO 70
150 NEXT K
160 MAT PRINT A
170 STOP
180 LET A(K.3)=0
190 GO TO 30
200 END
```

*Program III*

```
010 DIM A(6.3). B(20.3)
020 INPUT X
030 PRINT X;
040 IF X=1 THEN 150
050 LET Y=A(X.2)
060 LET Z=0
070 FOR K=1 TO 20
080 IF B(K.1) X THEN 130
090 LET N=Z+1
100 IF Y-A(N.2) B(K.2) THEN 130
110 LET X=N
120 GO TO 30
130 LET Z=Z+B(K.3)
140 NEXT K
150 END
```

### *Program IV*

```
010 DIM E(7.2), A(9.2)
020 MAT INPUT E. A
030 FOR K=1 TO 7
040 IF E(K.2) 0 THEN 140
050 LET E(K.2)=-1
060 PRINT K:
070 LET W=E(K.1)
080 IF E(K.1)=0 THEN 30
090 LET T=A(W.2)
100 LET E(T.2)=E(T.2)-1
110 IF A(W.1)=1 THEN 30
120LET W=W+1
130 GO TO 90
140 NEXT K
150 END
```

### *Program V*

```
010 DIM S(9.3)
020 MAT INPUT S
030 FOR L=1 TO 9
040 IF S(L.1)=1 OR S(L.2)=2 THEN 140
050 LET W=S(L.1)
060 LET T=0
070 FOR K=1 TO 9
080 IF S(K.2)  W THEN 120
090 LET P=S(K.3)
100 IF P  T THEN 120
110 LET T=P
120 NEXT K
130 LET S(L.3)=S(L.3)+T
140 NEXT L
150 LET A=3
160 LET T=0
170 FOR L=1 TO 9
180 IF S(L.2)  A THEN 220
190 LET P=S(L.3)
200 IF P   T THEN 220
210 LET T=P
220 NEXT L
230 PRINT"EVENT":A:"HAS VALUE":T
240 LET A=A+1
250 IF A=8  THEN 270
260 GO TO 160
270 END
```

# A Design for Computational Teaching Packages

**W. Y. Arms**
*The Open University*
*Milton Keynes, England*

*Correspondence address: Department of Mathematics*
*Dartmouth College*
*Hanover, New Hampshire 03755*
*(603) 646-2417*

## Abstract

This paper describes a simple but flexible overall design for on-line teaching packages where the subject being taught involves computation or data handling. The discussion includes the role of computing in the student's learning pattern, the overall design, important ergonomic details, and the instructions given to students. Seven packages based on this design have been written for an Open University course and are used as examples.

## Introduction

Successful instruction by computer involves more than simply providing a set of computer programs. For success the following four areas must be considered together: (1) Fitting the computing into the student's learning pattern, (2) the overall design of the computer packages, (3) attention to the details that make a system convenient to learn from, and (4) the instructions given to students.

This paper describes how computer instruction was fitted into an Open University course on numerical methods and techniques of operational research. The instruction is based on a simple overall design which has proved popular with students and tutors, and is suitable for many types of course.

## Open University Courses

Open University students are adults working part-time at home. Their main instruction is by printed texts, supported by television and radio broadcasts. They use computer terminals at about 200 locations scattered throughout Britain.

Preparing Open University courses presents a special challenge since the students usually work alone, and if anything is obscure they have nobody to turn to. Like most part-time students they are often pressed for time, and many have to travel significant distances to reach a computer terminal. If the first terminal session is unproductive or irrelevant to their needs they do not come back for another. To compensate for these difficulties the course teams are generously funded and are usually allowed two years to prepare each course.

## Computing in the Student's Learning Pattern

### *The Course*

The course teaches a variety of mathematical methods to honors level students who are not necessarily mathematics specialists. Each topic is introduced in a printed text, which contains calculator exercises but no computing. This is followed by practical work designed to consolidate the partially learned theory, not to teach new material. Television and radio broadcasts are used to clarify the more difficult concepts and to introduce small case studies. The following topics are covered: Solution of non linear equations, solution of systems of linear equations, linear and integer programming, discrete approximation, continuous function approximation, numerical integration, Monte Carlo integration, and simulation.

Several methods are introduced for each topic. For example three direct and three indirect methods are studied for solving systems of linear equations.

### A False Start

Our first idea for the practical work was to provide the students with a library of subroutines which they could incorporate into their own programs These subroutines could have been specially written or adapted from a standard library such as the Numerical Algorithms Group at Oxford.

We wasted considerable effort on this approach before realizing what should perhaps have been obvious. The students would spend their time writing routines to handle input and output while the methods that they were supposed to be studying were handed to them as black boxes.

### The Students' Learning Pattern

We should have begun by looking at the students' patterns of work. To explore the methods taught in the course, they are encouraged to follow the pattern in Figure 1.

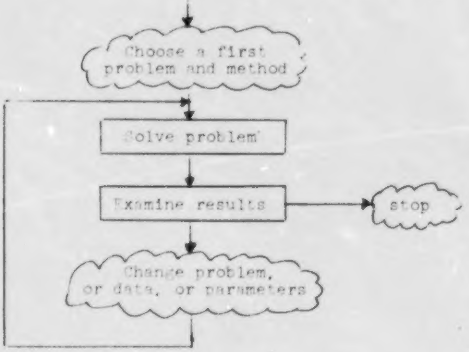

Choose a first problem and method
Solve problem
Examine results
stop
Change problem, or data, or parameters

**Figure 1. Pattern of Work**

The key feature is that the students solve a sequence of closely related problems. For example one of their first exercises is to explore how the smallest root of

$$\sin x + \frac{x-a}{x+1} = 0$$

varies with $a$. A student tackling this problem might tabulate the function for $a$ equal to 1, locate the root by bisection, change to a higher order method, try various starting values or stopping criteria and then repeat for a different value of $a$.

The practical work must encourage this sort of exploration. This requires computer packages which are easy to learn, flexible in use, have the minimum of input and print out a clear record of the work done for subsequent analysis at home.

## The Overall Design

A computer package has been written for each topic of the course. Except for the simulation, which is not discussed in this paper, these packages all have the same overall design and share many details. Figure 2 is the common outline flow chart.

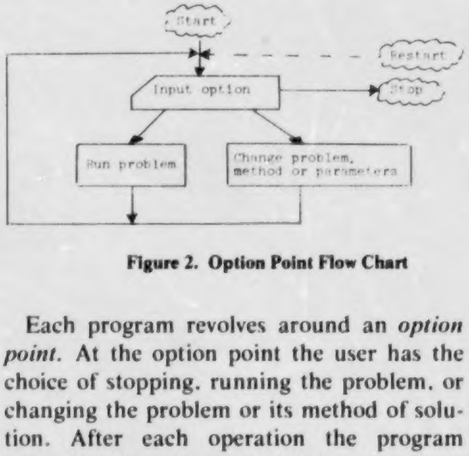


**Figure 2. Option Point Flow Chart**

Each program revolves around an *option point*. At the option point the user has the choice of stopping, running the problem, or changing the problem or its method of solution. After each operation the program returns to the option point. As an example of the kind of options offered, Figure 3 is a reproduction of a printed card which lists the options for the numerical integration package.

An overall design common to many packages is valuable to both the student and instructor. The student is spared unproductive time learning how to use a variety of programs. The instructor benefits because experience gained on one program can be used on the next, many sections of coding can be shared and pilot testing need not be carried out for each separate program. By building them on accumulated experience, the last two packages are in all ways better than the first two, yet required only a quarter of the effort to write.

$NCNI **Numerical Integration**

**Control Options**

0 Print the list of available options
1 Run the problem
2 Tabulate $f(x)$
8 List the problem
9 Stop

**Problem Options**

10 Input function
11 Input interval
12 Set function Parameter A

**Method Options**

20 Composite trapezoidal rule
21 Composite Simpson's rule
22 Composite Newton Cotes formula
23 Romberg method
24 Adaptive integration
25 Composite Gauss Legendre
26 Clenshaw Curtis method

**Parameter Options**

30 *n*
31 Order
32 Tabulation parameters
33 Stopping tolerance
34 Iteration limit *default* 20

**Print Options**

40 Full printout
41 Outline printout *default*

**Restart Procedure**

After BREAK or CTRL and C restart by typing

`RUN-100`

Figure 3. Sample List of Options

## Details

A good design can easily fail through lack of attention to details, to what can be called "program ergonomics," but detailed polishing is tedious and often rushed. Ideally the student's convenience and efficient use of the computer are less important. Here are some factors that we consider important and how they have been tackled.

### *Reducing Typing*

Computer instruction can easily turn into a test of typing. With the option point strategy no input need ever be repeated, whether it is data, a point option, or the specification of a method or its parameters. Flexible input and editing is provided, especially for data arrays. For example the linear and integer programming package offers options to input complete problems, either by specifying every element, or, for space problems, specifying only non zero elements. Individual elements of data can be edited, complete rows or columns added or deleted, constraint types changed, bounds set and variables made integer or non integer.

Input is also saved by the program structure. Since it is so simple no control instructions need be typed except the option codes. There are numbers rather than mnemonics, because numbers are shorter and unambiguous.

### *Terminal Output*

Open University terminals operate at ten characters a second. At this speed any redundant output is extremely tedious. Figure 4 is the printout from a short session looking for the root in (1,2) of the equation:

$$x^3-5x+3=0.$$

Data typed by the student are enclosed in boxes. Although the printout is brief, enough detail is given for it to be easily understood. For instance after each option code a short message is printed interpreting the code. One group of options allows the student to select the quantity of output, from a full printout at each step, to the solution only.

### *Error Trapping and Recovery*

Whatever errors the student makes are trapped so that he receives a comprehensible error message and does not lose his data. The example shows two simple errors that are easily trapped: an invalid option and not supplying necessary data. Arithmetic errors, such as division by zero, are more difficult. Their treatment is described in the next section. Finally the user is allowed to break into the program at any moment and restart at the option point as shown in Figure 2 without losing any data.

```
GET-$NCROOT
RUN
$NCROOT


OPTION ?20
TABULATE


OPTION ?10
F(X) = ?(X*X-5)*X+3


OPTION ?30
X0 = ?-3


OPTION ?31
X1 = ?3


OPTION ?32
NUMBER OF INTERVALS = ?12


OPTION ?1

TABULATE

 X                 F(X)
-3                -9
-2.5              -.125
-2                 5
-1.5               7.125
-1                 7
-.5                5.375
 0                 3
 .5                .625
 1                -1
 1.5              -1.125
 2                 1
 2.5               6.125
 3                 15


OPTION ?22
NEWTON-RAPHSON METHOD

OPTION ?1

NEWTON-RAPHSON METHOD
F'(X) NOT SPECIFIED
STOPPING TOLERANCE NOT SPECIFIED


OPTION ?11
F'(X) = ?3*X*X-5


OPTION ?33
TOLERANCE = ?.0001
```

Figure 4. A Typical Terminal Session

```
OPTION ?2.5
INVALID OPTION


OPTION ?30
X0 = ?2.5


OPTION ?1

NEWTON-RAPHSON METHOD

 N                 X                 F(X)
 0                 2.5               6.125
 1                 2.05455           1.39983
 2                 1.87188           .199559
 3                 1.83568           7.31325E-03
 4                 1.83425           1.09673E-05

SOLUTION
X = 1.83425
F(X) = 1.09673E-05


OPTION ?9

DONE
```

Figure 4 (continued)

### *Functions as Data*

Most computer languages require mathematical expressions to be included as language statements which are compiled. We have written two routines that allow expressions to be handled as data. The first checks an input expression for syntax errors and replaces it by a reverse Polish string. The second evaluates the string for given values of the variables. To trap errors, this second routine checks every arithmetic operation or function evaluation.

## Instructions to the Student

The student needs clear instruction on what to do before going to the terminal, what to take, and what to do at home afterwards. Since most terminals are in fairly cramped locations, we provide a printed card listing the options for each program. Figure 3 is an example. Option 0 of each package provides the same information.

Nobody ever reads documentation, but since the programs all have the same simple design only one set of documentation need be read. Full descriptions are provided, but once the basic idea has been understood students usually find all the information they need is provided by the printed card.

### Student Experience

The course was first presented in 1976 to over 500 students. A survey was made of their reactions to the practical work. Apart from complaints of work overload, the reactions have been very favourable. The programs have been easy to learn and many students have managed to carry out substantial assignments in a one hour terminal session. Perhaps the key indicator is that the enrollment for 1977 is much higher.

### Acknowledgment

Open University courses are produced by course teams. All members of the team contributed to the practical work of this course. In particular P. A. Blachford and F. L. Irvine were closely involved in the overall design. The packages were written mainly by F. L. Irvine and M. B. Haywood with help from the other members of the team.

# On the Development of a Proof

**Marialuisa N. McAllister**
*Moravian College*
*Bethlehem, Pennsylvania*

During the past few years there has been much debate on the use of the computer as an educational tool. The following classroom experience is given in support of the pedagogical value of the use of the computer. In the instances which are given in the sequel, it is the analysis of the numerical results which points out a significant pattern that leads the students to the development of an elementary proof.

A PDP 11/45 with 96K memory is available on the campus. Its operating system is Unix which we obtained from Bell Laboratories. This system supports about a dozen different languages and other utilities programs. Most of our use is timesharing with eight hardwired teletypes and four dial lines. It has a 300 line/minute printer, a 500 line/minute printer-plotter and 40 megabytes (i.e., 40 million characters) disk drives. We are also linked to the CDC 6400 of Lehigh University.

*On Infinite Series:* Most of the conventional tests which are introduced in a first calculus course fail to give an answer to the question of convergence of the series $\sum_{n=1}^{\infty} u_n$ when

$$u_n = \frac{n!e^n}{n^n}. \qquad (1)$$

The students are asked to evaluate a few terms of the sequence (1) by whatever means they choose. A helpful instructor might want to remind the students that the necessary condition for the convergence of a series is that $\lim_{n\to\infty} u_n = 0$. If the first 10 terms are evaluated with a HP-24 we have

| $n$ | $u_n$ |
|---|---|
| 1 | 2.718281828 |
| 2 | 3.694528051 |
| 3 | 4.463452648 |
| 4 | 5.118576568 |
| 5 | 5.699065322 |
| 6 | 6.225753017 |
| 7 | 6.711284214 |
| 8 | 7.164014889 |
| 9 | 7.589807965 |
| 10 | 7.992963906 |

The terms of the sequence (1) are positive with $u_1 = e$. The necessary condition for convergence does not seem to be satisfied because the evaluations point out that at least the first 10 terms are increasing. Is the sequence (1) increasing? If it is then it must be shown that for any $n$ we have $u_{n+1} > u_n$.
Indeed we have

$$\frac{(n+1)!\,e^{n+1}}{(n+1)^{n+1}} > \frac{n!\,e^n}{n^n}; \qquad (2)$$

$$\frac{(n+1)!\,e^{n+1}}{n!\,(n+1)\,e^n} > \frac{(1+n)^n}{n^n};$$

$$e > (1+\frac{1}{n})^n. \qquad (3)$$

Since the inequality (3) holds then the inequality (2) is true. Thus we have

$$u_1 = e < u_2 < \ldots < u_n < u_{n+1} < \ldots.$$

Its limit is not zero therefore the series diverges.

*On the rank of matrices:* Let $m \geq 2$ and let A be the matrix whose row vectors are given by

$v_i = ((i-1)m+1, (i-1)m+2, \ldots, im), \ i = 1, \ldots, m.$

It is easy to verify that for $m=3$ the matrix

$$\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix}$$

has rank equal to 2. The students are asked to experiment with $m$=4,5,6, etc., by using the matrix reduction routine. This routine is part of the linear algebra package which was created on our campus and is stored on the PDP 11/45 and on the CDC 6400. For example when $m=5$ we have

```
                         A IS SINGULAR,FIND THE RANK

                                      THE LAST REDUCED MATRIX:
       1.00000            .00000          -1.00000          -2.00000          -3.00000
        .00000           1.00000           2.00000           3.00000           4.00000
        .00000           -.00000           -.00000           -.00000           -.00000
        .00000           -.00000           -.00000           -.00000           -.00000
        .00000           -.00000           -.00000           -.00000           -.00000


ANY MORE JOBS FOR ME?n
```

These experiments suggest that the rank of A equals 2 for any $m$. Is this true? If a matrix has $m$ rows, $m>2$, and it has rank 2 then only two of its row vectors are linearly independent. Then it must be shown that there exist two real numbers $c_1$, $c_2$ such that

$$v_i = c_1v_1 + c_2v_2\,,\ i>2. \tag{4}$$

For $1\leq j\leq m$ the *j-th* components of $v_1$, $v_2$ and $v_i$ are given by $j$, $(m+j)$ and $(i-1)m+j$ respectively. If (4) holds then we have for all $j = 1, \ldots, m$

$$(i-1)m+j = c_1j + c_2(m+j)\,;$$

$$(i-1)m+j = c_2m + (c_1+c_2)m\,. \tag{5}$$

If the coefficients $c_1$, $c_2$ exist then they satisfy the system

$$\begin{cases} c_2 = i-1 \\ c_1 + c_2 = 1 \end{cases} \tag{6}$$

The solution of (6) is $c_1 = 2-i$, $c_2 = i-1$. Then the *j-th* component of $c_1v_1 + c_2v_2$ is

$$(2-1)j + (i-1)(m+j) = (i-1)m+j$$

which is precisely the *j-th* component of $v_i$. Thus (4) holds and A has rank 2 for any $m\geqslant 2$.

The computer has thus been used as a laboratory where experiments are performed. The analysis of the results, i.e., the computer evaluation, suggests the formulation of a conjecture and it hints at the method of proof. This approach is particularly successful with students with analytical minds. Students are constantly exposed to "do" mathematics. It is important whenever possible that they be exposed to the "making" of mathematics, namely to experiment and to formulate a conjecture which must then be proven.

# Computer-Oriented Calculus Courses Using Finite Differences[1]

**Sheldon P. Gordon**
*Department of Mathematics*
*Suffolk County Community College*
*Selden, New York 11784*
*(516) 233-5003*

## Introduction

At the Seventh Conference on Computers in the Undergraduate Curricula, the author presented a paper describing a new approach to teaching calculus. This so-called *discrete approach* involves introducing topics from the calculus of finite differences and finite sums both for motivation and as exceptionally useful tools for applications of the calculus. In particular, this approach provides an ideal setting in which to incorporate computers into calculus courses in an extremely natural way. The discrete approach and the class materials prepared by the author based on it have been used in several different calculus courses directed towards different audiences. They have been implemented within the traditional calculus sequence for math, science, and engineering majors at Suffolk County Community College and have been used in an applied calculus course for business majors at C. W. Post College.

The present paper is intended to report on the actual outcomes of these experiments. These detailed results and the accompanying analyses were not available in time for the previous paper, which consisted primarily of descriptions of the approach. In addition, the present paper will indicate how the discrete approach can be fruitfully used in virtually every type of calculus course offered today, not just in the non-science oriented course discussed in the previous paper.

As was pointed out in the earlier paper, there are a number of major benefits made possible by using the discrete approach. First, it provides a natural context for computer applications in the course, since all uses of the computer involve reducing continuous functions and operations to discrete analogues. Thus, the underlying mathematical concepts and techniques for computer usage will be available, rather than brought in in an often artificial way having little cohesiveness.

Second, having material on finite differences and sums available, it is a very simple matter to expand the scope of the course slightly to allow it to encompass several easy, yet useful, techniques from the field of numerical analysis. For example, approximation and interpolation methods can be covered which give the students a feeling for where all of those "given" functions really come from. In practice, no one will actually be solving significant problems involving calculus without the use of computers, and therefore an introduction to numerical methods is certainly desirable. All too few students go on to take a full course in the subject and, even for those who do, it usually stands as an isolated subject disjoint from the remainder of the mathematics courses taken in a typical undergraduate program. An early introduction to the subject within calculus should go far towards rectifying this situation.

Third, much of the current developments in applied mathematics are taking place in the non-physical sciences such as biology, economics, psychology, and so forth. In these areas, though, the majority of the mathematical models used are based on finite difference equations since the processes they reflect are inherently discrete. Consequently, the discrete approach provides the opportunity to develop some of these discrete models in a very natural way. For example, growth and decay problems are far easier to visualize in a discrete setting and, in fact, can be nicely covered during the first week of the course. This also provides some concrete application of the subject matter very early in the course, thereby sustaining student interest. This is especially necessary in an applied calculus course where the usual month or more of purely mathematical development is likely to lose the interest of most students.

Fourth, the students are exposed to most concepts and techniques twice, first in the discrete case and again in the continuous case. As a result, this on-going review provides the students with a greater understanding of and facility with the basic notions and methods of calculus. This is particularly beneficial in many calculus courses where ever greater numbers of students are entering with extremely poor backgrounds in mathematics.

### Detailed Description of the Approach

The mathematical foundations of the approach and the course are as follows. The calculus of finite differences and finite sums contains discrete analogues of most major concepts and techniques of the infinitesimal calculus without recourse to limits. For example, there are discrete parallels to such topics as maxima and minima of functions, the chain rule, the fundamental theorem of integration, techniques of integration, Taylor series expansions, and partial derivatives. (See, for example, any of the standard texts on the calculus of finite differences.) However, the absence of the limit reduces all work simply to the level of arithmetic and algebraic manipulation. Moreover, these discrete formulations are the natural ones to have available in any effort involving significant computer applications.

The idea behind the approach is to use the concepts and techniques from finite differences and sums as the underlying source of motivation for the infinitesimal calculus and the appropriate setting for solving "real" problems using discrete approximations. Specifically, most of the standard topics in calculus are taken up first in the discrete context. This leads to an easier and fuller understanding of the concepts and much greater facility with the techniques for most students in the class than is obtained in the purely continuous approach.

Furthermore, the infinitesimal calculus arises from the discrete calculus by simply passing to the limit as the step approaches zero. Thus, after the students have mastered a particular topic from the finite calculus and feel on firm ground, it is then a very simple and natural extension to introduce the limit process and thereby derive the corresponding result in the infinitesimal calculus. This natural transition has the additional advantage of presenting the students with a brief review of all concepts and methods as they are developed in the new context. Simultaneously, the students are shown how the discrete formulations can be used to approximate the continuous ones and they have the opportunity to actually apply all of these techniques on the college computer. This, in turn, fosters a still deeper understanding of the material.

The discrete approach produces a solid calculus course in which the mathematics is developed in a context which is directly oriented towards a computer philosophy. The use of the discrete foundation for the various topics provides the most natural setting for introducing the use of the computer for solving all types of problems involving calculus. Many of these areas where the computer has great usefulness, such as in evaluating limits, approximating derivatives, evaluating definite integrals, evaluating and approximating functions and so on, are already well known and need not be mentioned in any detail here. However, we will examine some of the particular applications which arise from the discrete approach and which are natural consequences of it. A course based on this approach begins with a study of sequences and the properties of the forward difference operator $\Delta$ acting on them. This is then extended to differences of an arbitrary function $y = f(x)$ defined on an interval $[a,b,]$. If a step $h$ is then chosen, the corresponding sequence of differences

$$\Delta f(a), \Delta f(a+h), \Delta f(a+2h), \ldots, \Delta f(a+nh)$$

can be easily determined, especially with the aid of a computer when the function is complicated or the step is small and the number of terms is correspondingly large. By analyzing the signs of these terms, it is possible to determine intervals where the function is basically increasing or decreasing and thereby find approximate locations for the maxima or minima of the function. For instance, if $f(x) = 8x^3 - 20x^2 + 14x - 3$ on the interval $[0,2]$ and if we choose $h = .25$, then we have the following table of differences:

| | |
|---|---|
| $f(0) = -.3$ | $\Delta f(0.) = 2.375$ |
| $f(.25) = -.625$ | $\Delta f(.25) = .625$ |
| $f(.5) = 0.$ | $\Delta f(.5) = -.375$ |
| $f(.75) = -.375$ | $\Delta f(.75) = -.625$ |
| $f(1.) = -1.$ | $\Delta f(1.) = -.125$ |
| $f(1.25) = -1.125$ | $\Delta f(1.25) = 1.125$ |
| $f(1.5) = 0.$ | $\Delta f(1.5) = 3.125$ |
| $f(1.75) = 3.125$ | $\Delta f(1.75) = 5.875$ |
| $f(2) = 9.$ | |

By examining these values, we see that the function is increasing overall between $x = 0$ and $x = .5$, then is decreasing overall from $x = .5$ to $x = 1.25$, and then is again increasing overall from there to the endpoint. Consequently, we (and the students) see that there is a maximum value for the function near $x = .5$ and a minumum value near $x = 1.25$. This type of development previews the entire concept of maximizing and minimizing functions at a stage prior to the introduction of the limit, let alone the derivative. Simultaneously, this approach sets the groundwork for the notion of a computer search operation for locating the approximate values of critical points. An added advantage of this early introduction of significant problems and the techniques needed to solve them is that the student is not required to spend the usual full month or more in purely mathematical development as in the traditional calculus course. Rather, he has enough mathematics available, using finite differences, to tackle interesting and/or practical problems and to have techniques available which are virtually begging for the use of the computer in their application. As a result, this approach enables the instructor to keep the students' interest at the early stages of the course where, in the traditional course in calculus, this is not the case.

Moreover, by using techniques such as these from the calculus of finite differences, it is possible to keep the course far more cohesive in the sense that this material presents a unifying theme for the course. In this regard, once the theoretical aspects of optimization theory are treated using derivatives, the above approach can be reintroduced for a variety of applications involving computerized solutions. For one, it can be applied to a given function to locate approximate values for critical points and these approximations are then used as initial values for the Newton's Method, say, to zero in on the roots.

Alternatively, the methods can be extended to supply discrete analogues of the first and second derivative tests for maxima and minima which can be applied to the entries in the difference table. For example, in the illustration above, we find that, using Newton's Method starting respectively at $x_0 = .5$ and at $x_0 = 1.25$, the critical points $f(x)$ are located at $x = .5$ and at $x = 1.1666666$. It is easy to see that the first difference changes from positive to negative about $x = .5$, thus indicating a probable maximum there. Similarly, about $x = 7/6$, $\Delta f$ changes from negative to positive, indicating a probable minimum at this point.

Alternatively, the second differences about $x = .5$ are negative, which also indicates a probable maximum there and similarly, about $x = 7/6$, $\Delta^2 f$ is positive, again indicating a probable minimum.

Admittedly, there are other numerical techniques which are faster, such as the bisection method to find the roots. However, the one described above fits into the calculus course in a very natural way and is also less likely to miss critical points if the step $h$ is taken small enough. Moreover, having had such an introduction to numerical methods, the student is much more likely to take a full course in numerical analysis subsequently and to realize the interactions between that subject and other areas of mathematics.

In terms of the actual implementation of the computer aspect of the course, the students were given several hours of programming instruction spread over about two weeks of classes. During the first use of the discrete approach, we used a simplified version of FORTRAN, while we used BASIC during a second trial. The latter is certainly preferable for students who have never had programming before. However, it is this writer's belief that FORTRAN would be more desirable in the sense that it is by far the more useful language in terms of eventual computer usage in practical situations.

Initially, to give the students some confidence and a gradual introduction to the language and its capacities, they were supplied with sample programs illustrating such concepts as evaluating limits of sequences and functions. They were expected to supply their own choices of sequences and functions and other parameters and to run these programs for a variety of different cases. Throughout the course, we supplied a series of sample programs, each successive one having more "holes" which the students had to fill. The following is a list of the topics covered in this manner:

1. Terms of a sequence.
2. Growth and decay.
3. Limit of a sequence.
4. $\varepsilon$-N limits of sequences.
5. Differences of functions.
6. Sequential limits of functions.
7. $\varepsilon$-$\delta$ limits of functions.
8. Derivatives from the definition.
9. Higher derivatives via limits.
10. Search for intervals of increase and decrease.
11. Newton's method.
12. Discrete tests for max and min.

## Analysis of the Results

The most significant component of this type of discussion regarding introducing additional material and techniques into an existing course must be the question of the time involved. In order to incorporate the material on both finite differences and the computer into the first term of the calculus sequence, it was necessary to devote approximately two weeks worth of class time, total. Normally, this would constitute an inordinate amount of the term. However, as indicated above, the discrete approach involves developing the usual calculus topics in the discrete case first and then carrying over in the limit to the continuous case. In each instance, we found that there is a very considerable saving in time during the development of the continuous cases, since the students were thoroughly familiar with the concepts already from the discrete setting. In fact, this is true to such an extent that the net loss in time was virtually nil. This was accompanied, though, by an overall major gain in the level of student understanding and facility, as measured by student performance on tests, student comments (both verbal and on course questionnaires) and my personal evaluation of individual classroom response.

In addition, the results of implementing the discrete approach at C. W. Post College bear this out fully. The approach was used in a one-semester applied calculus course for business majors. Considering the extreme shortness of such courses, the instructor chose to use the material solely as motivational without any direct use of the computer. As in our own experience, the loss of time involved in introducing the additional topics on finite differences was fully compensated for by the savings in developing the continuous cases. In fact, the instructor's personal opinion is that this material simplifed the calculus topics very considerably for the students and, as a result, of the ten sections which ran, this experimental section was one of the three that actually finished the entire syllabus. Most of the other sections did not have time to finish most if not all of the final chapter. (The book and syllabus have since been changed.) Moreover, despite introducing the extra material on finite differences and covering more calculus topics than most of the other sections, the overall grad average for the experimental section was virtually identical to the average department-wide, namely 2.55 versus 2.57 on a scale of 0 to 4.

In terms of the grades achieved in our experiments at Suffolk CC, complete figures are available for the spring, 1976 semester, but not yet for the fall, 1976 term. The results are summarized in the table below. The entry "No Credit" includes both failing students and those who have withdrawn during the course. The effect of combining the two drastically lowers the overall grade average in the course. Unfortunately, the breakdown is not available.

| | *Department-wide* | | *Experimental* | |
|---|---|---|---|---|
| | No. | % | No. | % |
| A | 28 | 12 | 8 | 26 |
| B | 43 | 19 | 3 | 10 |
| C | 48 | 21 | 7 | 23 |
| D | 26 | 11 | 4 | 13 |
| Incomplete | 4 | 2 | 1 | 3 |
| No credit | 77 | 34 | 7 | 23 |

The grade average for the 227 students enrolled department-wide was 1.60 on a scale of 0 to 4, while the results for the experimental section were 2.00. Thus, the results for the experimental section are significantly better than the department-wide figures. Admittedly, this can be attributed to a variety of causes, not the least of which is the particular group of students, which we personally feel was a bright group. However, we do feel that the approach used is also a major factor in the class' success, as is indicated in the following extract from a course evaluation we conducted:

1. To what extent have the presentation and the notes contributed to your interest in and liking for the course?

Greatly - 76 percent Little - 24 percent
Turned off -0

2. If you have taken calculus before, how would you rate the presentation using differences and computers?

Much better - 60 percent
Slightly better - 20 percent
Comparable - 20 percent
Worse - 0

3. What is the overall reaction to the materials distributed?

Excellent - 75 percent Good - 25 percent
Fair - 0 Poor - 0

4. How would you rate them compared to the text (Leithold's *Calculus*)?

Far superior - 73 percent Better - 27 percent
Comparable - 0 Worse - 0

As mentioned above, detailed results of the fall, 1976 semester are not available yet. The overall average for the experimental section, however, was about 1.7, and is therefore slightly better than the departmental results for the previous semester. Our assessment, though, is that this group was not an especially distinguished one.

In conclusion, we feel that the discrete approach is an extremely valuable one and can be profitably incorporated in virtually any calculus course, especially if there is to be a computer orientation. However, it can be used separately as purely motivational or for introducing discrete models, or any combination of these facets. Specific recommendations for its usage in a variety of different types of calculus courses are given in the following table.

| | *Comp.* | *Motiv.* | *Discr. Mods.* |
|---|---|---|---|
| Honors calculus | x | | |
| Traditional math/science calculus | x | x | x |
| Slow calculus | x | x | |
| One-term applied calculus | | x | x |
| Two term applied calculus | x | x | x |

## Note

1. The author gratefully acknowledges the support given through the State University of New York's program for Improvement of Undergraduate Instruction for the development of the project described.

## Reference

Gordon, Sheldon P. *A Computer Oriented Calculus Sequence for Social Science Students.* Proc. 7th Conf. on Computers in the Undergraduate Curricula, 243-246 (1976).

# A Method for the Use of Computing in a Conventional Calculus Course

**Dennis D. Berkey**
*Department of Mathematics*
*Boston University*
*264 Bay State Road*
*Boston, Massachusetts 02215*
*(617) 353-2560*

## Introduction

During the past decade the advantages of using the computer in teaching introductory calculus have been well-established, a good amount of relevant material has found its way into print ([12], [15], [16], [17], [9]) and the availability of computing facilities has become almost universal in this country. However, despite all these developments, the use of computers in introductory calculus instruction still has not found widespread adoption ([15]).

With respect to this question, the more radical departures seem to have received a majority of the attention, at least in the mathematical literature. The purpose of this article is to describe a way in which the use of computers can be successfully integrated into the confines of the traditional calculus course, and to report on one such implementation.

## Which Approach?

Many different models for the use of computers in calculus have been proposed, with none having been established as a clear preference.

The most radical approach has probably been that embodied by the Center for Research in College Instruction in Science and Mathematics (CRICASAM) project [8] which called for the integration of computing into a completely restructured calculus course. A textbook was produced which placed major emphasis on algorithmic concepts, resulting in a true computer-based introduction of the calculus. Although initially pursued with much interest and vigor, this approach has not found major adoption.

Another common approach has been to establish a separate course in parallel with the calculus, usually entitled *Calculus and Computing*, or something similar, in which the basics of programming are introduced and the students write programs involving topics from the elementary calculus. These courses usually carry one or two credits, and the student is expected either to have already completed some study in the calculus, or to be doing so concurrently. As such, these courses amount to an additional course basically concerned with the concepts of the elementary calculus.

In contrast to the above approaches, some calculus instructors have made use of the computer in a much more passive way. Either the computer has been used as a demonstration device (by simply displaying output or, in some cases by bringing a mini-computer into the classroom [11] or else instructors have written and "canned" programs which students simply execute at time-sharing terminals [7]. Along the same lines some institutions have used their time-sharing capabilities to develop self-help laboratories for calculus students ([1], [3], [5], [14]). These approaches have the advantage of minimizing the time spent on computer work, among other advantages, but do not involve the student in an active way with the programs being used.

It seems clear at this point that a wholesale changeover to a computer-based approach to the introductory calculus is not going to occur to any large extent. Moreover, in light of several related developments which have been occurring during the past decade, it is the opinion of this author that renewed attention should be given to the question of how best to integrate some nontrivial use of the computer into the traditional presentation of the elementary calculus without having to send students into separately designed laboratory or computer-calculus mini-courses.

First, students are now arriving much more sophisticated in the understanding and use of calculating devices than they were only a few years ago. Indeed, the advent of the inexpensive (sometimes programmable) hand calculator and the increased availability of computing facilities in the high schools have drastically closed the gap between the experience our students bring with them into the introductory mathematics courses and the understanding of the algorithmic methods necessary to successfully write and execute simple programs at a computing facility. (The article [9] contains an excellent up-to-date survey of the capabilities of hand calculators now generally available.)

Second, the accessibility of versatile high-level programming languages has significantly increased over the past decade. A few years ago, when one had to choose an initial programming language, a dichotomy existed between choosing a simple but limited instructional language (usually BASIC, or one locally produced) or a more versatile but cumbersome language such as FORTRAN for mathematical programming, or COBOL for business applications and manipulation of large data banks. With the development of PL/1 (and the PLC subset) one now has the opportunity of introducing students directly to a language which is versatile, yet simple to learn and use, and which they can continue to use should their interests in computing carry them beyond the type of programming done in their introductory calculus courses.

Third, and probably most important, is the rapid development of computer science as a separate discipline, in many ways distinct from mathematics. Indeed, undergraduate programs in computer science are well-developed at many institutions. Entry into most of these programs is not at the post-calculus point, but at the beginning of the freshman year. Moreover, the problems and goals set forth in these introductory courses often bear little semblence to the computational problems in calculus and numerical analysis through which we used to feel that we were introducing our students to "computing." (The "core" area of computer science education has been described in [13] as including the study of (1) programming languages, (2) operating systems, and (3) other subject matter concerned with the application of computing techniques to the study of computers.) Even at this author's institution, where the computer science programs, both undergraduate and graduate, have been developed wholly within the department of mathematics, it is not uncommon for an undergraduate to complete his studies with a concentration in computer science having been exposed to little of what is referred to as traditional mathematics, pure or applied. The point here is not a criticism of the emerging curricula in computer science—it is rather, to emphasize that in making use of the computer in the limited context of illustrating the implementation of the concepts of calculus we can no longer claim to be introducing students to computer science. That, for the most part, is something else.

In light of the above observations and the obvious fact that computers now play an important part in many of the related fields in which our students will eventually study and/or work (not to mention daily life) it is indeed desirable to make use of the computer in a nontrivial way in our calculus instruction. However, in doing so we need not feel obligated to devote any large amounts of time to introducing the "concepts of computing" or to make any significant changes in the traditional presentation of the calculus. Instead, we should simply introduce students to the facilities at hand, show them how to gain access to these facilities, and equip them with a few basic skills in writing the particular type of programs sufficient to carry out the calculations at hand. It is the opinion of this author that this can be accomplished within the confines of the traditional introductory calculus course, in minimal time, without having to resort to separate courses or tutorials for this purpose. For those students whose interest in computing extends beyond this limited framework, an introductory course in computer science is the logical next step. For all students, however, it should be made clear that the computer is an easily-accessible device which can be put to good use without a long period of training on the part of the user.

### The Experiment

To illustrate the author's claim that the computer can be successfully integrated into the traditional introductory calculus course the following experiment was conducted during the fall semester of 1976 at Boston University.

In an introductory course in the calculus, students were invited to participate in a programming project involving applications of the concepts to be studied during the semester. All work was to be done on the University's IBM 370/158 time sharing system whose interactive terminals are available at many locations across the campus.

One class period was initially devoted to an explanation of the computing facilities available to the students, the way by which one signs onto and off the computing system at a console, inputting and executing programs, and a very brief introduction to the PLC programming language. A sample program was presented which computed the mean of three numbers entered by the user via an interactive loop in the program. A short handout was distributed which described the procedures for signing on and off, and the use of the editing routine.

Once each week thereafter the students were given a one-page handout containing a printout of a program written by the instructor which performed a calculation relating to a topic currently under study in the course. In addition to the program the handout contained a set of instructions which typically asked the student to: (1) enter the program into his own program library, using the edit package to correct whatever mistakes were made during entry, (2) use the program to carry out several calculations (the programs were always written with an interactive loop which would ask the user to enter new data at the end of each calculation), (3) answer in writing several questions about what particular lines in the program were accomplishing, and (4) use the edit routine to modify the given program to one which would achieve a slightly different objective. (The objective, but not the particular modifications, was all that was given here.) About fifteen minutes of class time were devoted each week to a brief description of what the program was doing, new features not seen in previous programs, and student questions.

Thus, while the students received some initial information in writing each week (the sample program), the idea was that many of the basic ideas associated with the programs—looping, go-to statements, if statements, function routines—would be noticed and understood through interaction with working programs, rather than via the lecture method, as the complexity of the programs increased.

As reference material the students were advised to consult two publications available in the University bookstore. One [4] is a description of the University's computing facilities. The other [2] is an introductory text which happened to be in use in an introductory course in our computer science program.

The type of projects assigned were chosen from among the typical uses of the computer in illustrating the calculus—approximation of definite integrals, areas between curves and volumes, error analysis for Taylor series expansions, partial sums of infinite series, etc.—which have previously been well documented. The students were instructed to retain all printouts and to keep them, together with their written responses to the questions, in a notebook which was submitted at the end of the course.

Preparation time for these projects on the part of the instructor was found to be really minimal. Once a program was written and operating a listing was obtained on a ditto master. (For those who must work strictly with punched cards, a photo copy of the correct output could be used here.) Instructions were added, and the copies were run for distribution. The instructor, previously unfamiliar with the PL/1 language, found the text by Pollach and Sterling [10] an interesting and highly readable introduction to the language.

## Results

Response to the projects was very positive. Of the sixty-five students enrolled in the course, thirty-seven elected to participate in the projects.

At the end of the course a questionnaire was distributed to these students concerning the projects, to which twenty-seven submitted responses. The following subjective questions were asked:

1. Were the projects effective in illustrating the concepts from the calculus on which they were based?

2. Should the approach have been varied towards either the use of strictly "canned" programs which students would merely run but not enter or modify, or, at the opposite extreme, the requirement that students write all their own programs "from scratch"?

3. In what ways can the projects be improved?

In response to question one, twenty-five students said yes, two no. The affirmative replies included comments pointing out that the projects introduced the student to the available computing facilities, made the convergence concepts "come alive," illustrated the theory with "physical proof," and made it possible for students to "experiment with mathematics." One of the two negative replies was accompanied by the comment that the "computer cannot explain something" about which a student is already confused.

The response to question two constitutes the reason for which this article has been written. Of the twenty-seven students responding, none thought that canned programs should be used in the course. Twenty-four agreed with the approach that was actually taken, and three would have preferred to have written programs from scratch without the benefit of sample programs. Comments here pointed out that canned programs would not get the student as involved with the mathematics associated with the programs, and that writing all programs from scratch would have made too great a demand on both class and study time.

In response to question three, most students simply reaffirmed their satisfaction with the present approach. Several commented that they would like to have had more programs, a good number felt that the programs should be made mandatory of all students, and several made the suggestion that one or two of the last programs should be written from scratch by the students themselves.

The projects resulted in very little out of class demand on the instructor's time. The diagnostic features of the PL/1 language are quite extensive. Thus a student, armed with a sample of a correct program, can usually deduce the source of his difficulties from the diagnostic error messages and from a model of what he should have entered. Also, as is to be expected, several of the students had had some prior experience with computing. These students are generally delighted to practice their skills by helping classmates with their programming difficulties, and we suspect that most of the necessary assistance was obtained in this way.

## Conclusions

It is this author's belief that in most colleges and universities good use can be made of the available computing facilities by integrating their use into the traditional presentation of the introductory calculus. Students need to know that their computing facilities can now be approached in much the same way as their hand calculators—one simply needs to determine what the available facilities are, how to gain access to them, and what the rudiments of an appropriate language happen to be. Indeed, the cumbersome approach of separate courses or lengthy tutorials in programming have probably served to insure that some of our calculus students will take pains to avoid coming into contact with computing, since it has been made to appear so difficult.

As a final note it is interesting to observe the case of Dartmouth, where the concept of time-sharing was first introduced (1964) and where Kemeny produced the BASIC language. In the report [15] Stephen Garland notes that Dartmouth originally began with the supplemental approach to calculus and computing, then developed a more fully integrated approach, but has now come around to a state of more informal usage:

> ...students do as much or as little (programming) as they wish, with their activity ranging from using programs written by instructors to writing their own simple programs to developing a full-fledged interest in computing. Perhaps this last approach indicates that computing has finally come of age at Dartmouth; students and faculty expect to use computing just as they expect to use any other intellectual tool, but no longer feel compelled to call attention to the tool every time they use it.

## References

1. Caldwell, J. L. and Polley, Douglas, "Computer Assisted Tutorials in College Mathematics," American Mathematical Monthly 82 (1975) 846-848.
2. Conway and Gries, *An Introduction to Programming*, 2nd Edition, Winthrop, New York, 1975.
3. Garfunkel, Solomon, "A Laboratory and Computer Based Approach to Calculus," American Mathematical Monthly 79 (1972) 282-290.
4. Greiser, Tim, **Go—A RAX Primer*, Boston University Computer Center Publication.
5. Hethcote, H. W., and Schaffer, A. J., "A Computer Laboratory Course for Calculus and Linear Algebra," American Mathematical Monthly 79 (1972) 290-293.
6. Lax, Peter, Burnstein, Samuel, and Lax, Anneli, *Calculus with Applications and Computing*, Springer, New York, 1976.
7. Menzin, Margaret S., *Use of Canned Computer Programs in Freshman Calculus*, American Mathematical Monthly 83 (1976) 375-378.
8. Miles, E. P. Jr., "Calculus and the Computer, the CRICISAM Calculus Project—Past, Present and Portent," American Mathematical Monthly 78 (1971) 284-291.
9. Osofsky, Barbara, "Small Calculators for the Mathematician," Notices of the American Mathematical Society 23 (1976) 422-426.
10. Pollach and Sterling, *A Guide to PL/1*, 2nd Edition, Holt, New York, 1976.
11. O'Loughlin, Thomas, "Using the Electronic Programmable Calculator in Calculus Instruction," American Mathematical Monthly 83 (1976) 281-283.
12. Thesing, G. L. and Wood, C. A., "Using the Computer as a Discovery Tool in Calculus," American Mathematical Monthly 81 (1975) 163-168.
13. Wegner, Peter, "A View of Computer Science Education," American Mathematical Monthly 79 (1972) 168-179.
14. Williams, H. E. and DeBoer, Delmer, "Teaching a Computer-Oriented Laboratory Course for Ordinary Differential Equations," American Mathematical Monthly 80 (1973) 933-937.
15. "Calculus and Computing: The Influence of Computing on Mathematical Research and Education," Joint publication of the American Mathematical Society and the Mathematical Association of America, 1975.
16. "Calculus with Computers," report of CUPM, 1969.
17. "Recommendations on Undergraduate Mathematics Courses Involving Computing," report of CUPM, 1972.

# Calculus and Linear Algebra in A.P.L.[1]

**James W. England**
*Swarthmore College*
*Swarthmore, Pennsylvania 19081*
*(215) 545-7900 Ext. 251*

This paper is a report on an experimental calculus and linear algebra program which has been conducted during the last two years at Swarthmore College. The objective of the program is to teach calculus and linear algebra in a way that will make the ideas clear while not hiding the techniques and the problem-solving nature behind theorem-proof type courses.

To do this we have used computing in a central way: to represent the ideas and concepts of the subjects and facilitate their analysis and exploration. this is made possible by using the APL (a programming language) notation throughout the two courses. In doing so we have used the advantages of this notation to change the presentation of a number of the topics covered in these courses. The advantages we see in this notation are its lack of ambiguity, its ability to handle arrays in a straight forward manner, its relative conciseness, and its demand for explicitly defined functions. By using this notation we avoid the "translation problem." That is, by using APL we have one notation for expressing the concepts and techniques arising in the course and for directing a machine to implement these ideas (insofar as is possible). Another benefit of this notation is that it leads to a high degree of integration of linear algebra and calculus.

In these courses we stress the constructive and computational nature of the subjects. Many of the analytic tools, algorithms, and results of theorems are described as explicit functions. Once so defined they can easily be experimented with in order to gain a clearer understanding of their meanings and effect. They also remain available for use throughout the remainder of the course. Every function considered in these courses is introduced by at least one explicit construction. In addition to an understanding of the topics covered in the course we think the students leave the course with a rather different view of mathematics. This view includes experimentation. The demand for explicitly defined functions stresses a constructive view of these subjects and probably results in the students taking a similar view of mathematics. The nature of this program is one in which the notion of approximation is fundamental. Generally we proceed from a class of well understood functions to a new class of functions via approximation. In this way students see the idea of approximation used to define new objects as well as approximating known objects for computational purposes.

Polynomials are studied extensively in the calculus course. A polynomial is identified with its coefficient vector. Functions defined on polynomials are therefore defined on vectors. A large number of these functions are linear. This makes it useful to introduce the notion of linear functions and their matrix representation together with some elementary matrix algebra. The explicit representation functions on polynomials are used extensively in the discussion of Taylor's polynomials for transcendental functions.

The sections dealing with series are primarily devoted to power series. Series of numbers are considered only to check the convergence of power series at the endpoints of the interval of convergence. Topics in differentiation are covered in a rather standard way. However, since APL allows one to easily manipulate arrays, the students define a symbolic differentiator. That is, a function whose argument is a symbol string which represents an elementary function and whose result is a symbol string representing the derivative of the original function.

The following example may help explain some of the points which have been raised. The example is an outline of how part of the discussion of series can be presented. Early in the course we define a dyadic function *POLY* whose left argument is the coefficient vector of a polynomial and right argument is a vector consisting of the numbers at which the polynomial is to be evaluated. For example (¯4.0.1) *POLY*(¯2.0.2.4) results in (0.¯4.0.12). The polynomial associated with the vector (¯4.0.1) is. in standard notation ¯4+$X^2$. The function *POLY* allows us to make the identification between vectors and polynomials. A (power) series is a monadic function. *C*. with domain the positive integer such that 1.) (*N*+1) = ⍴ *C N*. and 2.) (*C K*) =*K*↑*C N* for each positive integer *N*. The discussion of convergence is done in terms of Cauchy sequences. The basic result is a statement of the "ratio test" in terms of Cauchy sequence of polynomials. To do this we need to define the function. *QT*. which results in the quotient of successive terms of a vector. An adequate definition of *QT V*. for *V* a vector. is (1↓*V*)÷−1↓*V*. (Of course we assume no element in *V* is zero.) Assume *C* is a series and notice that for *K*<*N*. |((*CK*) *POLY X*)−(*CN*) *POLY X* is equal to |((−⍴*C N*)↑*K*↓*C N*) *POLY X*.

*Theorem* For *C* a series. *B* a positive real number. and *N* and *K* are non-negative integers such that *K*<*N* define
M←B ⌈/QT K↓|C N. Then if 1>M
(|((−⍴C N)↑K↓C N) X POLY X)
≤(|(CN)[K]×B*K)÷1−M
for all real numbers *X* such that *B*≥|*K*. If we define a dyadic function *CEST* by *B CEST (K.N)* equal to (|(CN) [K]×B*K÷,−B×⌈|QT K↓|CN then *B CEST (K.N)* is an upper bound of the difference between the values of *CN POLY X* and *CK POLY X* on the interval ((−*B*)≤*X*)∧*X*≤*B*. Thus the problem of convergence of the series on this interval is the problem of showing that *B CEST (K.N)* gets small for large values of *K* and *N*. For example let *CN* be defined by ÷!⍳*N*+1. For *K*<*N*. QT K↓|CN equals ÷(K+1)+⍳(N−K+1). Thus ⌈/QTK↓|CN equals ÷K+1 and *B CEST (K.N)* is ( (K+1)×B*K)÷(K+1−B)×!K. Hence for *B*>0. *B CEST (K.N)* can be made arbitrarily small by choosing *K*<*N* large enough. Notice that for *B*=1.*B CEST (K.N)* is less than ÷!*K*−1. There are two ways this information is used. In some situations a (power) series is used to define a new function. Using the series of the above example we could define the exponential function as the limit of *(CN) POLY X*. In this case the information is used to show that the limit exists. and hence defines a function. In other situations the series *C* is defined as the Taylor's series of some known function. In this case the theorem provides an upper bound on the error made in using the value of a Taylor's polynomial of the given function for the value of the function.

The text we use is *Calculus in a New Key* by D. L. Orth. A.P.L. Press. Swarthmore. Pa. The above example is. in a different form. contained in Orth's Book.

Courses such as these. involving a rather radical departure from the standard courses in calculus and linear algebra. can create certain difficulties. The first and ultimately the most serious difficulty is its acceptance within the local scientific community. Calculus and linear algebra are subjects which are fundamental to the physical and social sciences. Hence any departure from the beaten path is suspect. At the same time. much elementary mathematics is learned by students as they attempt to use it in other science courses. Thus. there is a danger that students will not get reinforcement of their calculus and linear algebra in other courses. This problem seems to be one which can be decreased in its severity by a careful job of explaining the program to other departments. Also it should in time decrease. The precision of APL sometimes makes it difficult for people to use it informally. particularly in lectures. Our attitude in this regard has been to use the notation informally once it is understood by the students. Once again. this problem will lessen as people become more familiar with the APL notation.

Our experience with this program is such that we are attempting, whenever possible, to use the APL notation in other mathematics courses. In one sense this is an attempt to see to what extent APL can be used as a notation for mathematics. We hope that others will join us in doing this. Generally, the notation one uses for a subject tends to grow from the ideas in the subject. That is, topics considered important are easy to express while those considered less important become difficult to state. This probably accounts for the ad hoc nature of much mathematical notation. However, in trying to use APL as a notation for mathematics we are faced with the reverse situation. It is therefore important to remember that the notation one uses to describe a subject will shape one's view of the subject. Hence the use of APL as the notation for a subject usually requires one to completely rethink the subject. In calculus and linear algebra we think this has resulted in some important gains. This is the basis of our generally optimistic view of using APL as a notation for mathematics.

## Notes

1. Presented at the conference under the title "Calculus and Linear Algebra in APL."
2. Supported in part by NSF CAUSE grant 76-16071 to Swarthmore College.

# Computerized Applications of Matrix Algebra

**James T. Johnson**
*Mathematics Department*
*Dominican College of San Rafael*
*San Rafael, California 94901*
*(415) 457-4440 Ext. 234*

## Introduction

One important problem with which mathematics instructors at any level must contend is that of finding interesting applications of the mathematical concepts he or she is presenting. While professional mathematicians may find most mathematical topics of interest regardless of their usefulness, most undergraduates need a greater incentive to become genuinely interested in such topics.

This problem may be compounded for the instructor who intends to present the basic concepts of matrix algebra, whether it be to a college algebra, finite math, business math, linear algebra or computer programming class. While the trigonometry or calculus instructor can easily find a variety of applications of these subjects, it is often very challenging to find interesting applications of matrix algebra, especially matrix multiplication.

The fundamental concepts of matrix algebra are so readily accessible to undergraduates that it seems quite important that we develop and present applications of this important area of mathematics. This has been one of the author's areas of interest during his three years of teaching at Dominican College of San Rafael, a small (700 students), independent liberal arts college located about seventeen miles north of San Francisco.

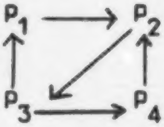
During the spring of 1974, Dominican entered the computer age by acquiring a teletype terminal connected via telephone to a computer at the Lawrence Hall of Science in Berkeley, California. That semester the author offered the first computer-oriented course, "Introduction to Matrices and Computer Programming." The objective was to present the essential concepts of matrix algebra and BASIC programming and to relate these two areas using applications. The text for the course was ***Linear Equations and Matrices*** by Johnston, Price and Van Vleck (Addison-Wesley, 1966).

Many of the applications presented were the obvious choices involving systems of linear equations, inventory matrices, and the like. However, one of the more interesting and successful applications involved communication matrices. Since first presenting this course this writer has repeated it once and given one-hour lectures on the topic of communication matrices to the Sonoma State College Math Colloquium (September 1975) and to the 1976 California State Mathematics Conference. The students and outside audiences have found the ideas of communication matrices quite interesting and understandable. Several teachers who attended these meetings have expressed gratitude at finding such a simple application of matrix algebra.

The purpose of this paper is to present some of the basic concepts involved in communication matrices and show how the computer can be used to facilitate the presentation of this topic, making it possible to study larger, more realistic systems.

## The Applications

Suppose we have a four point communication system with lines of communication occurring in the directions indicated in the diagram below.



This could represent an intercom system, police radios or four persons who do not have total communication with one another. We may be interested in some of the following questions:

1. To which point should a message be given so that it will get to every other point in the system in the least number of steps?
2. Suppose a message is given to a certain point. Will it eventually reach every other point? How many steps will it take?
3. Suppose that a message emanating from one point will never reach all of the other points. Can one or more additional lines of communication be installed to eliminate this problem? If so, where should these lines be installed?
4. On the other hand, can some lines of communication be eliminated and the same service maintained?
5. Which are the most important points in this system?

As a first step toward answering these questions, we form the communication matrix for this system:

$$M = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{vmatrix}$$

generated by $M_{ij}$

$$= \begin{cases} 1 \text{ if } P_i \text{ communicates directly with } P_j \\ 0 \text{ otherwise.} \end{cases}$$

For simplicity, $M_{ii} = 0$ for every $i$.

Clearly no point communicates directly with every other point, so it will take at least two steps to get a message to every point. The two step paths are:

$P_1$ to $P_2$ to $P_3$
$P_2$ to $P_3$ to $P_1$
$P_2$ to $P_3$ to $P_4$
$P_3$ to $P_1$ to $P_2$
$P_3$ to $P_4$ to $P_2$
$P_4$ to $P_2$ to $P_3$

and the two step communication matrix is:

$$A = \begin{vmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{vmatrix}$$

in which we place a 2 in the third row and second column since a message would get from $P_3$ to $P_2$ twice in two steps, once via $P_1$ and once via $P_4$.

If we compute the matrix $M^2 = M*M$ we find that

$$M^2 = \begin{vmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{vmatrix}$$

which is exactly the same as $A$, the two step communication matrix. This is not surprising if we examine the matrix $M^2$ carefully. The 1 in the first row and third column came from "multiplying" the first row and the third column of $M$. The 1 was achieved when the 1 in the first row and second column of $M$ was multiplied by the 1 in the second row and third column of $M$ was multiplied by the 1 in the second row and third column of $M$, as shown in the diagram below.

$$\begin{array}{ccc} \text{Row 1} & & \text{Column 3} \\ 0 & \longrightarrow & 0 \\ 1 & \longrightarrow & 1 \\ 0 & \longrightarrow & 0 \\ 0 & \longrightarrow & 0 \end{array}$$

$$0 + 1 + 0 + 0 = 1$$

But the 1 in the first row and second column of $M$ indicates a direct line of communication from $P_1$ to $P_2$, and the 1 in the second row and third column of $M$ indicates a direct line of communication from $P_2$ to $P_3$. Hence the 1 in the first row and third column of $M^2$ indicates a line of communication from $P_1$ to $P_3$ via $P_2$. A similar analysis can be made of each of the entries in $M^2$.

Furthermore, the matrix

$$M + M^2 = \begin{vmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{vmatrix}$$

indicates the lines of communication accomplished in one or two steps. From this matrix we can answer the first question. Either $P_2$ or $P_3$ would get a message to every other point in two steps or less, since the second and third rows have no zeros except on the main diagonal. We ignore zeros on the main diagonal since a nonzero entry there only indicates that a message sent from a certain point returns to the same point.

Similarly, the matrix

$$M^3 = \begin{vmatrix} 1 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 1 & 0 & 0 & 1 \end{vmatrix} \text{ indicates the three}$$

step communication paths, and the matrix

$$M+M^2+M^3 = \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{vmatrix} \text{indicates the lines}$$

of communication achieved in three steps or less. Since this matrix had no zero entry, we can see that a message given to any point in this communication system would eventually get to every point within three steps.

The general principle can be stated simply:

If we have a communication system of $n$ points $P_1, P_2, \ldots P_n$ we first form the communication matrix $M$ defined by:

$$M_{ij} = \begin{cases} 1 \text{ if } P_i \text{ communicates directly with } P_j \\ 0 \text{ otherwise.} \end{cases}$$

Then, for any positive integer $K$, $M^k{}_{ij}$ is the number of paths of communication from $P_i$ to $P_j$ in exactly $k$ steps, and $(M + M^2 + \ldots + M^k)_{ij}$ is the number of paths of communication from $P_i$ to $P_j$ in $k$ steps or less. Note that since there are only $n$ points, it could never take more than $n-1$ steps to communicate between any two points. If two points do not communicate within $n-1$ steps, they will never communicate. Hence $M^{n-1}$ and $M + M^2 + \ldots + M^{n-1}$ are the last two matrices that would ever need to be computed.

Answering the third and fourth questions can be quite difficult. Clearly what we need to do is experiment with adding and deleting lines of communication. This can be done easily by changing 1's to 0's and vice versa in the matrix. However, the arithmetic, especially in a large system, becomes unbearable. This is where computer programming skills become necessary.

The following program is the essential tool needed to answer the questions stated above. After inputting the size of the communication system and the original communication matrix, this program causes the computer to print out the matrices $M$, $M + M^2$, $M + M^2 + M^3$, $\ldots$, $M + M^2 + M^3 + \ldots + M^{n-1}$. The user may then answer the first two questions by inspection and run the program several times, changing the original matrix, to try to answer the third and fourth questions.

```
10 Input "How many points in the communication
  system? ". N
 20 Print
 30 Dim M(N.N). B(N.N). C(N.N). D(N.N)
 40 Print "Enter the communication matrix"
 50 Mat Input M
 60 Print "1 Step"
 70 Mat Print M;
 80 Mat B=M
 90 Mat C=M
100 for I=2 to N-1
110 Mat D=M*B
120 Mat B=D
130 Mat C=C+B
140 Print I; "Steps or Less"
150 Mat Print C;
160 Next I
```

Several of my students have written variations of this program to answer specific questions. For example, the following program is used to answer the first question directly. It determines the point(s) to which a message should be given so that it will get to all of the other points in the least number of steps.

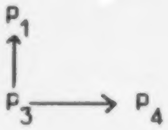
```
 10 Let P=0
 20 Input "How many points in the communication
  system? ".N
 30 Print
 40 Dim M(N.N). B(N.N). C(N.N). D(N.N)
 50 Mat Input M
 60 Mat B=M
 70 Mat C=M
 80 for S=1 to N-1
 90 for I=1 to N
100 for J=1 to N
110 if I=J then 130
120 if C(I.J) = 0 then 160
```

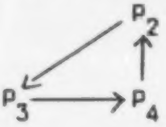
```
130 next J
140 Let P=1
150 Print "Point"; I; "In"; S; "Steps or Less"
160 Next I
170 if P=1 then End
180 Mat D=M*B
190 Mat B=D
200 Mat C=C+B
210 Next S
220 Print "No total communication point exists"
```

Answering the fifth question, as to which points are most important, is, of course, a matter of some personal preference. However, it is possible to define what is known as a liaison point.

A communication system is said to be connected if there is a path of communication between any two points in the system. If the system has $n$ points and if $M$ is its original communication matrix, then the system is connected if and only if the matrix $M+M^2+\ldots+M^{n-1}$ has no zero entry except possibly on the main diagonal.

A point is called a liaison point if its removal from the system results in a disconnected system. In the example above, $P_2$ is a liaison point since its removal results in the disconnected system shown below. This system is disconnected since $P_1$ and $P_4$ never communicate and $P_1$ would never receive any messages. On the other hand, $P_1$ is not a liaison point since its removal results in the system shown below which is still connected.





In a general $n$ point communication system with original communication matrix $M$, it is easy to remove a point from the system by setting all the entries in the corresponding row and column equal to zero. Then one can test to see if the resulting system is disconnected by computing $M+M^2+M^3+\ldots+M^{n-1}$ and checking for zeros except on the main diagonal and on the row and column corresponding to the point being considered. Again the arithmetic becomes unbearable since this process must be repeated for each of the $n$ points. Clearly a computer program is needed for this task.

The following program, a revision of one written as a class project by one of my students, determines which points in a system are liaison points.

```
 10 Rem Program to find liaison points
 20 Input "How many points in the communication
  system? ",N
 30 Print
 40 Dim M(N,N), A(N,N), B(N,N), C(N,N), D(N,N)
 50 Print "Enter the communication matrix"
 60 Mat input M
 70 Print
 80 for K=1 to N
 90 Mat A=M
100 for I=1 to N
110 for J=1 to N
120 if I=K then A(I,J) = 0
130 if J=K then A(I,J) = 0
140 Next J
150 Next I
160 Mat B=A
170 Mat C=A
180 for T=2 to N-1
190 Mat D=A*B
200 Mat B=D
210 Mat C=C+B
220 next T
230 for I=1 to N
240 for J=1 to N
250 if I=K then 300
260 if J=K then 290
270 if I=J then 290
280 if C(I,J) = 0 then 330
290 next J
300 next I
310 Print "Point":K:" is not a liaison point"
320 go to 340
330 print "Point":K:" is a liaison point"
340 next K
```

The topic of communication matrices is vast indeed. The principles outlined above can be used to examine many types of communication systems and associated problems. The interested reader should consult the text by Johnston, Price and Van Vleck cited earlier for more information on these topics. Such a venture will surely provide the math instructor with many interesting ways of using the computer to augment the study of matrix algebra and its applications.

# Teaching a Survey of Mathematics for College Students Using a Programming Language

**Edward J. LeCuyer, Jr.**
*Western New England College*
*1215 Wilbraham Road*
*Springfield, Massachusetts 01119*
*(413) 783-6131*

## Introduction

This manuscript is a description of work done by the author during the academic year 1975-1976, while he was a doctoral student at the School of Education, University of Massachusetts.

The investigation has two major components: (1) A textbook, written by the author, entitled *A Survey of Mathematics for College Students Using a Programming Language*, and (2) an exploratory study into the use of A.P.L. in teaching a survey of mathematics course to college students. It represents one response to the C.U.P.M. plea for experimentation in teaching basic college mathematics using the computer.

In the textbook, a survey of basic topics in mathematics (topics usually covered in a freshman survey course) is presented using A.P.L. notation together with conventional mathematical notation. That is, concepts and algorithms are described using symbols from both notations, usually in parallel. Discussions of applications of all topics are presented, followed by exercises and problems. Computer programs presented in the text are intended to enable the student to observe and experiment with the mathematical concepts and algorithms, and programs for computer solution (i.e., programming projects) are included as optional exercises. This textbook consists of the following chapters: Chapter 1: Set Theory; Chapter 2: Logic; Chapter 3: Vectors and Matrices; Chapter 4: Systems of Linear Equations; Chapter 5: Determinants; Chapter 6: Functions and Graphs; Chapter 7: Exponential and Logarithmic Functions; Chapter 8: Differential Calculus; Chapter 9: Integral Calculus; Chapter 10: Probability; Chapter 11: Statistics; Chapter 12: The Trigonometric Functions; and Appendix.

Using this textbook, an exploratory study was conceived in order to investigate the effectiveness of using A.P.L. and the computer in teaching a survey of mathematics. This study was conducted at the University of Massachusetts in the spring of 1976 for two sections of Math 100: Mathematics in the Modern World, with eighteen students and twenty-seven students (mostly freshmen) respectively. One section was taught using A.P.L., access to a computer terminal, and the above mentioned textbook. The other section was taught the same topics without A.P.L. or the computer using a comparable textbook.[1]

Math 100 is the lowest numbered mathematics course at the University of Massachusetts and is usually taken by students who want three credits in science-mathematics as part of an area requirement. These people normally do not take any further mathematics courses and typically are not very interested in mathematics. Many of them have had difficulty with mathematics in the past. There are no prerequisites for this course, and many students usually have a weak background in mathematics. The students in the author's classes consisted primarily of freshman liberal arts students. On the average, they had a background consisting of from one to one and a half years of high school algebra with an average grade of

C-. Their average college board score in mathematics was in the low 400's. Thus, it could be said that these students represent a challenge to the instructor.

Considering the caliber of student in Math 100, and the fact that the study took place over only one semester, only a subset of the topics in the textbook could be covered. Specifically, the following chapters of the author's textbook were covered in the following order: Chapter 1: Set Theory; Chapter 6: Functions and Graphing; Chapter 7: Exponential and Logarithmic Functions; Chapter 11: Statistics; Chapter 3: Vectors and Matrices; and Chapter 4: Systems of Linear Equations.[2]

Both classes were administered a multiple-choice, diagnostic pretest constructed by the author, and designed to measure their knowledge of the topics to be covered before the start of the course. At the end of each set of two chapters, the classes were given parallel forms of a multiple-choice test prepared by the author. Finally, at the conclusion of the course, both classes were given a final examination, which was actually the same as the pretest. The author attempted to design the tests so that they would measure the understanding of concepts, rather than just computational skills and memory. In fact, on all tests after the pretest, students were allowed to use a one-page reference page of notes which they themselves wrote (sort of a "legal cheat sheet"). This eliminated the need for the students to memorize. The author also attempted to design these tests so that they would favor neither class. The results of these tests are summarized in the table below:

| | *Mean for the A.P.L. Class (18 students)* | *Mean for the Non-A.P.L. Class (27 students)* |
|---|---|---|
| Pretest | 3.39 | 4.37 |
| Test 1 on Set theory and Functions | 17.11 | 17.0 |
| Test 2 on Exponential and Logarithmic Functions and Statistics | 18.22 | 16.40 |
| Test 3 on Matrices and Systems of Linear Equations | 19.6 | 18.9 |
| Final Exams | 18.5 | 19.04 |

As one can see from this table, there was no significant difference between the performances of the two classes on these tests.

In addition to these tests, both classes were given a questionnaire to determine their initial attitudes towards themselves as mathematics students, toward mathematics, and toward the computer. At the end of the course, they were given a second questionnaire designed to measure changes in attitude toward the course, mathematics, and the computer as a result of the course. The questionnaire also attempted to measure final attitudes toward the course, the instructor, the text, and the computer.

Based on the results of these questionnaires, it could be said that the A.P.L. class enjoyed using A.P.L. and the computer to learn mathematics. The students did not find this approach exceedingly difficult, and many felt that the use of A.P.L. and the computer made the course seem more interesting and relevant to them. Many also felt that the use of A.P.L. and the computer helped them to better understand the mathematics. In general, the A.P.L. students reacted very favorably to the use of A.P.L. notation and the computer in learning mathematics.

## Implications of the Study

Since this is only a small, one-shot exploratory study and not a large formal experiment, no globally conclusive statements can be made. However, the author does feel that the evidence of this study supports the following implications. These implications merit further study, and the author hopes that further research will be undertaken in each of these areas.

1. Students with no previous background in computer languages or computer usage can be taught A.P.L. along with mathematics.

In general, the students in this exploratory study had little difficulty in learning A.P.L.

and conventional mathematical notations simultaneously. They were able to use A.P.L. and the computer to do their homework. The same amount of mathematics was taught in the usual time, indicating that A.P.L. can be taught along with mathematics without sacrificing very much time from the mathematics.

2. Students have at least as good an understanding of the concepts of mathematics when they learn mathematics using A.P.L. as they do when they learn it the conventional way.

The results of the tests in this exploratory study seem to support this statement. In general, the A.P.L. students did at least as well on these tests as the non-A.P.L. students. Also, many of the A.P.L. students themselves stated that they felt that A.P.L. and the computer helped them to understand the concepts in the course.

3. Students using the A.P.L. approach learn more than students using the conventional approach.

The test results indicate that the A.P.L. students learned at least as much mathematics as the non-A.P.L. students. In addition, they learned a versatile skill—computer programming—and they learned to use a powerful tool—the computer. They learned A.P.L., a very powerful programming language. They learned these computer-related skills without sacrificing the learning of mathematical skills and concepts. Thus, the A.P.L. students learned more than the non-A.P.L. students.

4. The use of A.P.L. and the computer does not detract from the students' ability to do computations.

Since the author does not feel that computational ability is a very important facet of a course such as Math 100, he did not explore this issue thoroughly. However based upon the test questions which required computational ability, it does not appear that the A.P.L. students were significantly less able to perform mathematical computations than the non-A.P.L. students, despite the fact that much of their computations on the homework problems wre done by the computer.

5. Many students enjoy learning mathematics using A.P.L. and the computer.

In general, most of the A.P.L. students reacted favorably to this method of learning mathematics. While there were a few people who indicated that they did not like this method, they were more than offset by people who indicated that they liked this approach. For some people, the use of the computer made a significant difference in their enjoyment of mathematics.

6. Incorporating A.P.L. and the computer in the basic mathematics course is a good way to get students who would not otherwise use the computer to become familiar with this important tool of our modern world.

Many people indicated that they would never have used the computer if it were not for this course. This is true in spite of the fact that on the initial questionnaire they stated that they believe that our present world is largely influenced by computers and that they believe that everyone should know something about the use of the computer. After having used the computer in this course, they indicated that it was not as hard to use as they had imagined it would be, that they enjoyed it, that it made the course more interesting, and that they would use the computer in the future if the need arises. They also indicated that they had a better attitude toward computers at the end of the course than they had in the beginning.

7. The author's textbook can be used effectively to teach a survey of mathematics course using A.P.L.

The author feels that the results of this exploratory study support this statement. However, it should again be pointed out that the author of the text was also the instructor in the exploratory study.

## General Recommendations

The author would like to make some recommendations to teachers of college survey of mathematics courses.

1. Use the computer in teaching a college survey of mathematics course.

The computer is rapidly becoming an extremely important and useful tool in our modern world. We are all becoming more and more affected by computers. Therefore, it would seem to be important that all college graduates know something about using the computer. However, many college students will not take a computer course per se. An excellent way of acquainting these people with the computer is by incorporating the computer in the basic mathematics course. This can be done in such a way that the amount of mathematics learned will not be decreased. In fact, by incorporating the computer in the mathematics course, the student may learn the mathematics better and enjoy it more.

2. Use A.P.L. in teaching a college survey of mathematics course.

A.P.L. is an easy to learn, yet extremely powerful and useful computer programming language. It was first developed as a notation for doing mathematics and can be taught very effectively along with the mathematics.

3. Assuming the teacher agrees to use A.P.L. and the computer in teaching his survey of mathematics course, he must then be willing to accommodate his role as a teacher to fit these new, powerful teaching tools.

It may become necessary for the teacher to make some changes in his teaching role and technique in order for the students to benefit fully from the use of the computer and A.P.L. in the mathematics course. With this in mind, the author would like to make some specific suggestions to teachers who would like to use an approach similar to the exploratory study described briefly in this paper. These suggestions are based on the author's experiences in this study.

### Specific Suggestions

1. Use A.P.L. notation and conventional mathematical notation in parallel.

This is a new approach to teaching mathematics. However, based on this exploratory study, the author feels it is a good one. Having to translate between the two notations will help strengthen the mathematical concepts for the students. The two notations are compatible enough so that the students will not be confused at learning two notations simultaneously.

2. If possible, have a computer terminal directly available in the classroom.

The author found it rewarding to have a different student seated at the terminal in each class. In this way, the class can see first hand the ease with which mathematics can be done at an A.P.L. keyboard. The student programmer becomes a key part of the class. This makes the rest of the class more alert and active. If a student asks a "what if" question, the student programmer can try out the question on the terminal and get an immediate answer, thereby encouraging experimentation with mathematics. Before any proofs are done in class, several specific examples can be done by a member of the class at the terminal, thereby giving the class confidence in the fact being proved. The computer can do computations very quickly and the class time will be saved because of it.

3. If possible, have a computer terminal available for the students to do their homework outside of class.

The biggest criticism the students in this exploratory study had of using the computer to learn mathematics was of the nonaccessibility of a computer terminal for doing homework. The demand for terminals was great, and the students would sometimes have to wait a long time to get a turn at the terminal. For some students, this inconvenience and waiting time overshadowed the advantages of using the computer to do their homework. If possible, reserve a centrally-located terminal especially for the students in your class.

4. If possible, the instructor or a student assistant should be available to assist the students at the terminal.

This is especially important at the beginning of the course, when the students are more likely to have difficulties at the terminal. In the beginning, the author's students made several errors, and if the instructor had not made himself available at the terminal, some of these students would have become frustrated and might have given up. After the students have obtained a feel for the terminal and have learned to correct their errors, it will not always be necessary to have an assistant available at the terminal.

5. Strongly encourage the students to try to use the terminal at the beginning of the course.

Getting the students to take that first step at the terminal is very important. Once the students have tried using the terminal and see that it isn't very difficult, they'll be "hooked." However, getting them to take that first step is sometimes difficult. Perhaps the instructor could use the first class or two to have the students take turns experimenting at the terminal. The author found this approach very valuable in getting students over their initial fear and reluctance to use the terminal.

6. Collect homework and write helpful comments on it.

Perhaps this is standard and should be included in any mathematics course. The fact that the instructor is interested enough in the students' homework to read it carefully will lead most students to do a conscientious job on it. It is important that the student take an active role in this course. He learns mathematics by doing it. The student will always do the homework in a course in which it is collected before he will do it in a course where it is not. This is especially true if comments are made on the paper. These comments may be quite helpful to the student in detecting and correcting errors. If there are no errors, the comment "nice job!" will be appreciated by the student.

7. In a computer oriented class, be a little patient with the students about their homework.

Because of difficulties in getting access to a computer terminal and possible difficulties at the terminal, it may not be possible for all students to do the homework immediately. Of course, they should be encouraged not to let the assignments pile up. However, a little tolerance on the time should be allowed.

8. Avoid making students type in long programs.

When typing in long programs, many students will make errors, even if they type in the program successfully, they will become concerned about the amount of time needed to type in the program before they can use it to do the homework. If possible, the instructor could store these longer programs in a common workspace which would be available to the class. However, if the program is short, the student should be encouraged to type it in himself. One wouldn't want the student to view the computer as merely a source of canned programs.

9. With regard to programs, this author would recommend presenting the student with the completed program, plus an explanation of the program and some illustrations of its use.

The emphasis is put on how to use the existing program to do the mathematics involved. The student may be asked to write some programs of his own in some of the exercises. However, in general, he should be provided with programs needed as a foundation for the development of the mathematics.

While the author realizes that by providing already written programs he is perhaps depriving the student of the opportunity to develop the algorithm for doing the particular mathematical operation himself, and of the opportunity to write, debug, and rewrite his own program, it is his belief that the above approach is the appropriate one for the Math 100 type student. The program should be explained so that the student fully understands what it is doing for him. However, the emphasis is put on the mathematics and not on the programming.

10. Finally, this author recommends that the instructor concentrate on mathematics in class.

Don't use class time for discussing programming debugging techniques, etc. Let the students read the appendix or a programming manual and have someone available at the terminal to help the students overcome programming difficulties. Keep in mind that this is a mathematics class and not a programming class. Emphasize the power and usefulness of A.P.L. as a mathematical notation and tool. Take advantage of this power and usefulness, but don't get distracted by programming details. Introduce the A.P.L. only as needed in the development of the mathematics.

In conclusion, this author found the approach of using A.P.L. and the computer to teach a survey of mathematics to be a rewarding and enjoyable one, and would recommend this approach to others who teach a survey of mathematics to college freshmen. Hopefully, the textbook *A Survey of Mathematics for College Students Using a Programming Language* will be available during 1977 for those in need of a texbook for teaching such a course.

## Notes

1. *Mathematics: With Applications in the Management, Natural, and Social Sciences,* by Margaret L. Lial and Charles D. Miller, Scott Foresman Co., 1974.

2. This order of topics was chosen because it corresponds to the order of topics in Lial and Miller's text.

## References

*Recommendations on Undergraduate Courses Involving Computing*—October 1972 by the C.U.P.M. of the Mathematical Association of America.

Birkhoff, Garrett. "The Impact of Computers on Undergraduate Mathematical Education in 1984," *The American Mathematical Monthly*, June-July 1972.

Falkoff, A. D. and Iverson, K. E. "The Design of A.P.L.," *The A.P.L. Quote Quad*, Spring 1975.

Iverson, Kenneth E. *A Programming Language*. John Wiley and Sons, New York, 1962.

Iverson, Kenneth E. "The Use of Computers in Teaching Mathematics," *IBM Technical Report* #320-3014, July 1972.

Peelle, Howard A. *Instructional Applications of Computers Using A Programming Language*. Addison-Wesley, Reading, Mass., 1976.

# Computers and Calculators: A Strategy for Combined Usage in Trigonometry

**Samuel W. Spero**
*Department of Mathematics*
*Metro Campus*
*Cuyahoga Community College*
*2900 Community College Avenue*
*Cleveland, Ohio 44115*
*(216) 241-5966*

## Introduction

One of the important uses of the computer in mathematics instruction at Cuyahoga Community College is for generating problem sets for students. A different collection of problems is prepared for each student. The students work on these problems either at home or in class. Depending on what the instructor has in mind these problem sets may be used for drill or for testing and evaluation. A crucial feature of this application of the computer is that at the same time that the computer generates the problem set for the student, it also generates the solution set for the instructor who is then able to evaluate each student individually.

In this article we will present the instructional strategy which we use in incorporating both the computer and the calculator into our trigonometry courses. The computer is used to generate problem sets for the students and solution sets for the instructor. The students are required to use calculators to complete their individually assigned problem sets. Among the results achieved by the use of this instructional strategy, we have found increased student motivation, increased achievement with respect to objectives in trigonometry, and the acquisition of considerable skill in the use of personal "scientific" calculators by the students.

## College Trigonometry

The trigonometry course under discussion is the traditional course in which are enrolled students in engineering-oriented terminal vocational programs.

We were led to attempt our experiment with computer-generated problem sets as a result of an ongoing problem in mathematics instruction. That is the problem of evaluation. The usual approach to evaluation is to administer a single examination to the entire class during a regular class meeting. The difficulty in using this approach is that in areas of mathematics which are computationally-bound, the instructor must limit himself to the most trivial of problems so that the student can complete the examination within a single class meeting. This is especially true in a number of areas studied in trigonometry.

An alternative is to assign the students examinations which they work on at home. This approach is very appealing in that no class time is lost and the instructor can assign nontrivial problems. The difficulty associated with take-home examinations is that you cannot be sure that students will not work together.

Of course if each student is assigned a different problem or problem set on which to work, the difficulty is resolved—except for the instructor who now has to prepare many more problem and solution sets.

At this point it occurred to us that we might use the computer to generate both the problem and solution sets. Using the random number generator implicit in the BASIC computer language, it is not difficult to generate as many different problem sets as needed. We can even "personalize" each

196

problem set by adding the student's name. This feature also helps identify the relevant solution set when the problem sets are graded.

A further advantage presented itself almost immediately. If for one reason or another, a student is unable to take the test at the regularly scheduled time, there is no difficulty in preparing a "make-up" examination. In fact, if a student has difficulty in completing the particular problem set assigned to him, we will help him and then assign a new problem set. Since our ultimate objective is student learning this seems most reasonable.

At the outset of the experiment we were concerned whether our rather limited computer facility would be able to accommodate us. Our mathematics department owns a Data General Nova 1220 minicomputer with 16K words of core storage. There is no mass storage device attached to the computer, and the computer serves three terminals simultaneously. We use the BASIC computer language, and it is core resident. Thus our programs must be relatively short. Programs are saved on paper tapes which can be read into the computer at ten characters per second via the paper tape readers attached to each of the teletypes. As will be seen shortly the programs which generate the problem sets and their solutions are not long. We have not had any difficulty in accommodating them to our limited facility.

An objection that we anticipated from the students was that the non-trivial problems generated by the computer were much too tedious to solve by hand. So we decided to "let them eat cake."

For the past year or two we have noticed increasingly that the students enrolled in the trigonometry course own "scientific" calculators. In addition, we noticed that they do not know how to use them effectively. It seemed to us a waste not to have the students take advantage of this tool to help them learn more trigonometry, especially since they are always asking us if they can use their calculators on examinations, in preparing homework assignments, etc. If helping the students learn to use their calculators more efficiently helps them learn more trigonometry, and at the same time permits the instructors to use computer-generated problem sets, it seemed reasonable to integrate the calculator into the curriculum. This we have now done, which brings us to the topic of this article.

### Computer Programs

Before describing two of the areas in which we have used the computer/calculator strategy, we will describe the strategy itself. In all of the cases to be described the material to be learned is first presented using the textbook for the course.[1] Students are encouraged to use their calculators throughout the course, but the exercises in the text itself do not demand calculator usage. The computer/calculator strategy begins at the conclusion of the unit being discussed. The strategy is basically a strategy for evaluation.

First a typical problem is generated by the computer. This we solve in class by way of review of the material discussed in that unit of study. This also provides the students with the opportunity to work out the problem-solving algorithms in terms of their own calculators. Several computer generated problems are then assigned to the entire class as a homework (non-credit) assignment. These are subsequently discussed during class and all questions concerning the solution of these problems are answered. Finally, each student is given his or her assignment to be completed at home. Each student is informed that if he has any difficulty completing the problem set he can get help and will be assigned another problem set. In all of these assignments we attempt to remove the threat of failure.

Angles, Their Measure and Their Trigonometric Functions: This is the very first unit in the course. Among the objectives in this unit are an understanding of the two ways in which we measure angles—degree and radian measure—and how to convert from one system to the other. In this unit the students also learn the definitions of the trigonometric functions and their interrelationships, as well as the concept of the inverse trigonometric functions. We want students to understand the concept of a circular function and how a given angle, no matter how large, may be reduced to an angle less than 360°. Finally we want students to understand that any angle can be related to an angle in the first quadrant in terms of the absolute value of their respective trigonometric functions.

In order to evaluate students with respect to all of the objectives just described, we have the computer generate three types of problems. In problem-type I the computer generates a large angle measured in degrees. The student must convert that angle to radians. He must also relate the given angle to an angle less than 360° (which we call the related angle) and convert it to radians. He must then determine the reference angle which is the angle less than 90° that has the same trigonometric functions in absolute value. The reference angle must also be specified in both degrees and radians. Finally, for the given angle (or related angle) the student must specify all of the six trigonometric functions. Sample student output for this type of problem is included as Figure 1. Note the student's name in the output.

The type II problem is identical in all respects except that the angle given to the student is given to him in radian measure. A sample of the output for the student is included as Figure 2

**Figure 1**

```
PROBLEM FOR    SAM SPERO

GIVEN AN ANGLE OF                 1790.78       DEGREES:

FILL IN THE MISSING VALUES:

ANGLE                             FUNCTIONS
GIVEN ANGLE
DEGREES        RADIANS            SIN           CSC

RELATED ANGLE
DEGREES        RADIANS            COS           SEC

REFERENCE ANGLE
DEGREES        RADIANS            TAN           COT
```

**Figure 2**

```
PROBLEM FOR    SAM SPERO

GIVEN AN ANGLE OF                 32.0408       RADIANS:

FILL IN THE MISSING VALUES:

ANGLE                             FUNCTIONS
GIVEN ANGLE
DEGREES        RADIANS            SIN           CSC

RELATED ANGLE
DEGREES        RADIANS            COS           SEC

REFERENCE ANGLE
DEGREES        RADIANS            TAN           COT
```

Problem type III is a bit different. In this case the student is given two of the trigonometric functions of the angle. To simplify the computer program we have specified the two functions. These are the cosecant function and the tangent function. These two functions serve to uniquely define the angle. This angle will be less than 360°— what we call the related angle. Using the inverse trigonometric functions we want the student to determine the related angle, both in degrees and radians. We also want the student to determine the corresponding reference angle in degrees and radians. Finally, we want the student to determine the original angle generated by the computer which is equal to the related angle plus some multiple of 360 (in degree measure). To this end we give the student the value of the multiple of 360° which is needed to determine the given angle. This angle must also be determined both in degree and radian measure. A sample of the output for the student is included as Figure 3.

In each of the problem types the computer also generates for the instructor the student's name, the information given to the student, and the solution. For example the instructor's output for a type III problem would be as shown in Figure 4.

Figure 3

```
PROBLEM FOR     SAM SPERO

GIVEN THE COSECANT FUNCTION =  1.00829
      THE TANGENT FUNCTION =   7.7516
      THE GIVEN ANGLE = THE RELATED ANGLE +   18  X 360:

FILL IN THE MISSING VALUES:

ANGLE                          FUNCTIONS
GIVEN ANGLE
DEGREES         RADIANS        SIN            CSC

RELATED ANGLE
DEGREES         RADIANS        COS            SEC

REFERENCE ANGLE
DEGREES         RADIANS        TAN            COT
```

Figure 4

```
SOLUTION FOR    SAM SPERO
GIVEN THE COSECANT FUNCTION =  1.00829
      THE TANGENT FUNCTION =   7.7516
      THE GIVEN ANGLE = THE RELATED ANGLE +   18  X 360:
ANGLE                          FUNCTIONS
GIVEN ANGLE
DEGREES         RADIANS        SIN            CSC
 1522.65         26.5752        .991781        1.00829
RELATED ANGLE
DEGREES         RADIANS        COS            SEC
 82.6504         1.44252       .127945        7.81584
REFERENCE ANGLE
DEGREES         RADIANS        TAN            COT
 82.6492         1.4425         7.7516         .129006
```

As can be seen, the role of the calculator in these exercises is considerable. In problem types I and II the students must first use their calculators to convert from radian measure to degree measure or vice versa. Depending on the type of calculator they own this may involve arithmetic, or it may involve no more than pressing a single key. What is crucial is that the student master his own calculator. Next he must be able to reduce the given angle to a related angle and then to a reference angle. This involves straightforward, but tedious, application of principles studied in the unit. The calculator allows this aspect of the problem to be completed quickly. The student must recognize the trigonometric function keys on his calculator and how to use them to determine the sine, cosine, and tangent functions. He must also recognize the importance of the reciprocal key which he uses to determine the cosecant, secant, and cotangent functions.

In problem type III the student must use the inverse trigonometric function keys. He must also understand the material studied, because the inverse function keys return only an angle between -90° and +90°. This represents the reference angle. The student must perform an analysis on the two given functions to determine in which quadrant the related angle lies. Then using the calculator and the reference angle the student can determine first the related angle and finally the given angle.

As described above, we first solve each of the problem types in class, using calculators. Then several are assigned for homework from each of the problem types. Finally each student is given one of each problem type for evaluation purposes.

The computer program for the above is included as Appendix 1.

### Applications of the Trigonometric Functions—the Triangle Problem

The classical application of the trigonometric functions which all students study is the use of the law of sines and the law of cosines in the analysis of the parts of a triangle and in the calculation of its area. The objectives in this unit of study include the definitions of the trigonometric functions in terms of a triangle, the law of sines, and the law of cosines. A number of theorems from elementary geometry are also crucial to this unit. These include the Pythagorean Theorem and the theorem which states that the sum of the angles of a triangle must be 180°. In addition the students must learn to analyze which of four possible cases has been presented to them, and then determine the appropriate problem-solving algorithm.

In the triangle problem the students must determine all six parts of a triangle. These include the three angles and the three associated sides. In the problems given to the students they are presented with three of the six parts. Using the theorems and the trigonometric laws they have learned, they have sufficient information to analyze the triangle completely. The four cases are as follows:

1. Given: three sides.
2. Given: two sides and the included angle.
3. Given: two sides and the angle opposite one of them.
4. Given: two angles and one side.

Once the students have determined all of the parts of the triangle, the students determine the area using the known parts of the triangle.

In the course of the discussion of this unit, the students learn to apply the law of sines and the law of cosines using their calculators. Each student develops in his notes the appropriate button-pushing sequence required for his own calculator. Thus by the end of the unit, the students have developed all of the tools they require to solve the given problems quickly once they have performed the appropriate analysis.

In class we present examples of the problem types which might be encountered. Similarly, for homework the students receive examples of all types for practice. In the evaluation exercise, however, the students receive several problems which are randomly generated. They may be the same type or entirely different. The analysis is a major aspect of the evaluation.

Figure 5 is a sample of the output from the computer. It includes both the student output and the instructor output. The instructor must, of course, separate the two sections. this approach was necessary from the point of view of saving time in the production of the problems, since we were limited to the output from a ten-character-per-second teletype. It also permitted us to use a much simpler programming procedure which was more conducive to the limitations of our hardware.

```
SOLVING TRIANGLES


      *
     * *
    * Z *
  A*     *B
  *       *
 * Y     X *
*************
     C


ANGLE OPPOSITE SIDE  A  IS ANGLE  X
ANGLE OPPOSITE SIDE  B  IS ANGLE  Y
ANGLE OPPOSITE SIDE  C  IS ANGLE  Z


STUDENT NAME -- SAM SPERO         STUDENT NAME -- SAM SPERO


PROBLEM  1                        PROBLEM  1 --TYPE  1
GIVEN:  ANGLE X=  3.63262         GIVEN:  ANGLE X=  3.63262
        ANGLE Y=  45.2274                 ANGLE Y=  45.2274
        SIDE  C=  92.9515                 SIDE  C=  92.9515
SOLVE THE TRIANGLE
        ANGLE Z=                          ANGLE Z=  131.14
        SIDE  A=                          SIDE  A=  7.82
        SIDE  B=                          SIDE  B=  87.62

THE AREA IS                       THE AREA IS  258.01


PROBLEM  2                        PROBLEM  2 --TYPE  2
GIVEN:  ANGLE Z=  28.67           GIVEN:  ANGLE Z=  28.67
        SIDE  A=  37.71                   SIDE  A=  37.71
        SIDE  B=  80.01                   SIDE  B=  80.01
SOLVE THE TRIANGLE
        ANGLE X=                          ANGLE X=  21.0848
        ANGLE Y=                          ANGLE Y=  130.245
        SIDE  C=                          SIDE  C=  50.2903

THE AREA IS                       THE AREA IS  723.766
```

```
PROBLEM  3
GIVEN:  ANGLE Y=   28.5179
        SIDE  B=   37.84
        SIDE  A=   44.15
SOLVE THE TRIANGLE
        ANGLE X=
        ANGLE Z=
        SIDE  C=

THE AREA IS
```

```
PROBLEM  3 --TYPE  3
GIVEN;  ANGLE Y=   28.5179
        SIDE  B=   37.84
        SIDE  A=   44.15

        ANGLE X=   33.8521
        ANGLE Z=   117.63
        SIDE  C=   70.2185

THE AREA IS  740.06
```

```
PROBLEM  4
GIVEN:  SIDE  A=   63.18
        SIDE  B=   95.43
        SIDE  C=   95.7639
SOLVE THE TRIANGLE
        ANGLE X=
        ANGLE Y=
        ANGLE Z=

THE AREA IS
```

```
PROBLEM  4 --TYPE  4
GIVEN:  SIDE  A=   63.18
        SIDE  B=   95.43
        SIDE  C=   95.7639

        ANGLE X=   38.5917
        ANGLE Y=   70.4183
        ANGLE Z=   70.99

THE AREA IS  2850.22
```

**Figure 5**

The computer program for the above is included as Appendix II.

Other areas in the trigonometry course where we use this approach are in graphing the sine and cosine functions for arbitrary amplitude, frequency and phase angle and in multiplying, dividing, and exponentiating complex numbers. Those interested may obtain the computer programs and sample output from the author.

### Summary and Conclusions

The instructional strategy we have presented in this article—which includes the use of the computer to generate problem sets for students and solution sets for teachers as well as extensive student use of calculators to solve the problem sets—has had a number of ramifications beyond the original intent.

The original intent of this approach was to permit more effective evaluation of student learning. While we have not tested this approach formally (a research project is presently underway); intuitively we feel that the approach does provide more effective evaluation. Not only are we able to provide students with their own individual problem sets which they can work at home, but also we have been better able to crystalize our objectives for the students. By being able to focus directly on the learning objectives in instruction, the students are more aware of what is expected of them. This leads to improved learning.

In addition, encouraging students to use their calculators has increased student motivation considerably. Students spend time developing effective algorithms for solving problems with their calculators. As a result they have learned more trigonometry. Thus we have not only developed a tool for evaluation, but we have been able as a result to motivate our students and to help them learn more.

In using the computer to generate problem sets for students and solution sets for the teacher, we feel that we have touched upon an approach which places the computer in a role for which it is unquestionably well-suited. In this role it provides the instructor with technological assistance in completing a task which he could not reasonably perform himself. To provide this assistance we require only a minimal computer facility which makes the computer's assistance cost effective. The instructor, on the other hand, reaps maximum benefit. Using the computer in this way we have been able to motivate the student; we have been able to help him learn concepts more effectively, and we have been able to evaluate the effectiveness of the entire process. What more can we ask of computer-assisted-instruction?

## Note

1. I. Drooyan, W. Hadel, C. Carico, *Essentials of Trigonometry*, MacMillan Publishing Co., N.Y., 1971.

## Appendix 1

### Program for: Angles: Their Measure and Their Trigonometric Functions

```
100 READ S$
110 DATA "SAM SPERO"
120 PRINT"PROBLEM FOR  ",S$
130 PRINT
140 PRINT
150 LET D=RND(-1)*100
160 LET S=SIN(D)
170 LET C=COS(D)
180 LET T=S/C
190 LET S1=1/S
200 LET C1=1/C
210 LET T1=1/T
220 LET D1=D*180/3.14159
230 LET M=INT(D/6.28318)
240 LET D2=D-M*6.28318
250 LET D4=ABS(ATN(T))
260 LET D3=D2*180/3.14159
270 LET D5=D4*180/3.14159
280 LET N=INT((D+.005)/D2)
290 LET B=INT(RND(-1)*3)+1
300
310 ON B GOTO 340,420,500
320 PRINT"SOLUTION FOR",S$
330 ON B GOTO 380, 460, 560
340 PRINT"GIVEN AN ANGLE OF",D1,"DEGREES:"
350 PRINT
360 GOSUB 620
370 GOTO 320
380 PRINT"GIVEN AN ANGLE OF", D1,"DEGREES:"
390 PRINT
400 GOSUB 770
410 GOTO 940
420 PRINT"GIVEN AN ANGLE OF",D,"RADIANS:"
430 PRINT
440 GOSUB 620
450 GOTO 320
460 PRINT "GIVEN AN ANGLE OF",D,"RADIANS"
470 PRINT
480 GOSUB 770
490 GOTO 940
500 PRINT"GIVEN THE COSECANT FUNCTION =",S1
510 PRINT TAB(6);"THE TANGENT FUNCTION =",T
520 PRINT TAB(6);"THE GIVEN ANGLE = THE RELATED ANGLE + ";M;" X 360:"
530 PRINT
540 GOSUB 620
550 GOTO 320
560 PRINT"GIVEN THE COSECANT FUNCTION =",S1
570 PRINT TAB(6);"THE TANGENT FUNCTION =",T
```

```
580 PRINT TAB(6);"THE GIVEN ANGLE = THE RELATED ANGLE + ";M;" X 360:"
590 PRINT
600 GOSUB 770
610 GOTO 940
620 PRINT"FILL IN THE MISSING VALUES:"
630 PRINT
640 PRINT"ANGLE",,"FUNCTIONS"
650 PRINT"GIVEN ANGLE"
660 PRINT"DEGREES","RADIANS","SIN","CSC"
670 PRINT
680 PRINT"RELATED ANGLE"
690 PRINT"DEGREES","RADIANS","COS","SEC"
700 PRINT
710 PRINT"REFERENCE ANGLE"
720 PRINT"DEGREES","RADIANS","TAN","COT"
730 PRINT
740 PRINT
750 PRINT"--------------------------------------------------------"
760 RETURN
770 PRINT
780 PRINT
790 PRINT"ANGLE",,"FUNCTIONS"
800 PRINT"GIVEN ANGLE"
810 PRINT"DEGREES","RADIANS","SIN","CSC"
820 PRINT D1,D,S,S1
830 PRINT
840 PRINT"RELATED ANGLE"
850 PRINT"DEGREES","RADIANS","COS","SEC"
860 PRINT D3,D2,C,C1
870 PRINT
880 PRINT"REFERENCE ANGLE"
890 PRINT"DEGREES","RADIANS","TAN","COT"
900 PRINT D5,D4,T,T1
910 PRINT
920 PRINT
930 RETURN
940 END
```

## Appendix 2

## Program for: Applications of the Trigonometric Functions—the Triangle

```
100 PRINT
110 PRINT "SOLVING TRIANGLES"
120 PRINT
130 PRINT
140 PRINT  TAB(8);"*"
150 PRINT  TAB(7);"* *"
160 PRINT  TAB(6);"* Z *"
170 PRINT  TAB(4);"A*     *B"
180 PRINT  TAB(4)"*       *"
190 PRINT  TAB(3);"* Y     X *"
200 PRINT  TAB(2);"*************"
210 PRINT  TAB(7);"C"
220 PRINT
230 PRINT "ANGLE OPPOSITE SIDE  A  IS ANGLE  X"
240 PRINT "ANGLE OPPOSITE SIDE  B  IS ANGLE  Y"
250 PRINT "ANGLE OPPOSITE SIDE  C  IS ANGLE  Z"
260 PRINT
270 PRINT
280 READ N$
290 DATA "SAM SPERO"
```

```
300 PRINT "STUDENT NAME -- ";N$; TAB(40);"STUDENT NAME -- ";N$
310 LET B=0
320 FOR J= 1 TO 4
330 FOR I= 4 TO 5
340 LET A(I)=INT(RND(-1)*10000)/100
350 IF A(I)<1 THEN 340
360 NEXT I
370 LET A(3)=INT(RND(-1)*16000)/100
380 IF A(3)<10 THEN 370
390 LET T1=A(3)*3.14159/180
400 LET A(6)=SQR(A(4)↑2+A(5)↑2-2*A(4)*A(5)*COS(T1))
410 LET T2=(A(5)↑2+A(6)↑2-A(4)↑2)/(2*A(5)*A(6))
420 LET T2=SQR(1-T2↑2)/T2
430 IF T2>=0 THEN 460
440 LET A(1)=180-ABS(ATN(T2))*180/3.14159
450 GOTO 470
460 LET A(1)=ATN(T2)*180/3.14159
470LET A(2)=180 -A(1)-A(3)
480 LET B=B+1
490 PRINT
500 PRINT
510 PRINT"PROBLEM ";J;TAB(40);"PROBLEM ";J;"--TYPE ";B
520  ON B GOTO 530, 610, 690, 770
530 PRINT"GIVEN: ANGLE X= ";A(1);TAB(40);"GIVEN: ANGLE X= ";A(1)
540 PRINT TAB(7);"ANGLE Y= ";A(2);TAB(40);TAB(47);"ANGLE Y= ";A(2)
550 PRINT TAB(7);"SIDE  C= ";A(6);TAB(47);"SIDE  C= ";A(6)
560 PRINT"SOLVE THE TRIANGLE"
570 PRINT TAB(7);"ANGLE Z= ";TAB(47);"ANGLE Z= ";A(3)
580 PRINT TAB(7);"SIDE  A= ";TAB(47);"SIDE  A= ";A(4)
590 PRINT TAB(7);"SIDE  B= ";TAB(47);"SIDE  B= ";A(5)
600 GOTO 850
610 PRINT"GIVEN: ANGLE Z= ";A(3);TAB(40);"GIVEN: ANGLE Z= ";A(3)
620 PRINT TAB(7);"SIDE  A= ";A(4);TAB(47);"SIDE  A= ";A(4)
630 PRINT TAB(7);"SIDE  B= ";A(5);TAB(47);"SIDE  B= ";A(5)
640 PRINT"SOLVE THE TRIANGLE"
650 PRINT TAB(7);"ANGLE X= ";TAB(47);"ANGLE X= ";A(1)
660  PRINTTAB(7);"ANGLE Y= ";TAB(47);"ANGLE Y= ";A(2)
670 PRINT TAB(7);"SIDE  C= ";TAB(47);"SIDE  C= ";A(6)
680  GOTO 850
690 PRINT"GIVEN: ANGLE Y= ";A(2);TAB(40);"GIVEN; ANGLE Y= ";A(2)
700 PRINT TAB(7);"SIDE  B= ";A(5);TAB(47);"SIDE  B= ";A(5)
710 PRINT TAB(7);"SIDE  A= ";A(4);TAB(47);"SIDE  A= ";A(4)
720 PRINT"SOLVE THE TRIANGLE"
730 PRINT TAB(7);"ANGLE X= ";TAB(47);"ANGLE X= ";A(1)
740 PRINT TAB(7);"ANGLE Z= ";TAB(47);"ANGLE Z= ";A(3)
750 PRINT TAB(7);"SIDE  C= ";TAB(47);"SIDE  C= ";A(6)
760 GOTO 850
770 PRINT"GIVEN: SIDE  A= ";A(4);TAB(40);"GIVEN: SIDE  A= ";A(4)
780 PRINT TAB(7);"SIDE  B= ";A(5);TAB(47);"SIDE  B= ";A(5)
790 PRINT TAB(7);"SIDE  C= ";A(6);TAB(47);"SIDE  C= ";A(6)
800 PRINT"SOLVE THE TRIANGLE"
810 PRINT TAB(7);"ANGLE X= ";TAB(47);"ANGLE X= ";A(1)
820 PRINT TAB(7);"ANGLE Y= ";TAB(47);"ANGLE Y= ";A(2)
830 PRINT TAB(7);"ANGLE Z= ";TAB(47);"ANGLE Z= ";A(3)
840 GOTO 850
850 PRINT
860 LET T3=.5*A(5)*A(6)*SIN(A(1)*3.14159/180)
870 PRINT"THE AREA IS";TAB(40);"THE AREA IS ";T3
880 NEXT J
890 END
```

# Computer Exercises in Geophysics

**Paul M. Doherty**
*Physics Department*
*Oakland University*
*Rochester, Michigan 48063*
*(313) 377-3420*

### Introduction

The recent fuel shortages and growing public awareness of the quality of the environment have awakened the interest of a generation of students in the important problems of location and exploitation of raw material and energy resources. As a result, university enrollments are increasing in earth science areas. As students study the earth, they find that the tools of geophysics seem to have been extracted from a physics textbook. For example, a geophysicist extends his human senses with gravity and magnetic sensors, he looks through the earth with sound waves, and he measures heat flows so small that a snake would be envious. The computer is an important tool in analyzing this vast array of data. In line with the above trends, undergraduate and graduate programs in geology are offering more courses in geophysics, and at Oakland University we are using the computer as an integral part of our geophysics courses.

This paper describes the use of computers in an undergraduate geophysics course at Oakland University. Computer use is based on a student manual titled, "Computer Exercises in Geophysics," which is designed to accompany many standard geophysics textbooks. The manual contains exercises which the students work using prepared BASIC language programs on time-shared terminals. There is no computer programming prerequisite for the course, yet we find that students rapidly learn the BASIC language. Most of the exercises require that the students understand the operation of a program, and a few exercises ask them to write or modify programs. Student response to the use of the computer has been positive.

The computer serves a dual role in this course. Not only does it serve as a powerful teaching tool, but also it acquaints the student with the techniques of computer analysis, which are important tools of the geophysicist. The computer is used for teaching in two ways: (1) students use prepared programs to work exercises, and (2) a graphics terminal is used as a teaching aid during recitation periods. The techniques of computer analysis are taught by having the students study the basic techniques used in the prepared programs.

### Contents of the Student Manual

The application of the laws of physics to solve problems in the real world is often difficult, even though the laws can be expressed as one-line equations. In geophysics, one often encounters problems which do not have closed-form solutions or which require the use of tabulated data. In these circumstances, it becomes advantageous to use computers to solve the problems. By providing the student with programs that solve geophysical problems and then asking him to work with and understand these programs, he will gain insights into the geophysics as well as gaining valuable experience in techniques of computer solution. In addition to problems in which a numerical solution is necessary, there are instances in which the computer can be used to produce a graphical output which can be a great aid to understanding.

The student manual contains examples of all of the above uses of the computer. Some programs solve problems which *require* computer solution, such as: finding the gravitational anomaly of complex bodies in three dimensions, calculating the shape of a rotating self-gravitating planet, calculating the path of seismic waves in the earth, using Monte-Carlo techniques to find possible density distributions in the earth, solving Laplace's and Poisson's equations for time-independent heat flow in the earth, and solving the heat diffusion equation in two-dimensions. All of the above programs produce graphical output as described below. In addition, the manual also contains exercises to build student intuition. For example, there are programs which trace magnetic field lines of dipoles and quadrupoles, and which teach Fermat's principle. In addition, several exercises use geophysics examples to teach computer techniques. In particular, computer integration is illustrated by examining the percentage of the earth's surface beneath a given altitude, and by calculating the central moment of a planet.

The student manual is integrated into the course in the following manner. First, the background physics material is presented in lecture. Usually this material is illustrated by problems which are capable of solution by hand, this might include an easy integral. Then the students are given an assignment in which the computer has been programmed to solve a more realistic and usually more difficult example. The computer exercise has the following form: the student is given an initial set of conditions for the program which he runs. Then he is directed to vary these input parameters, predict the output based on his understanding of the physics, and finally compare his prediction with the computer solution. He thus learns by using the scientific method: observe, hypothesize, and test. At this point, the students are encouraged to make up their own initial conditions. When the exercise has been completed, they present their discoveries to the professor and their classmates in a recitation section. A portable graphics terminal is used in the recitation sessions for these presentations. The interaction between the student, the professor, and the computer can lead the students, and occasionally the professor, to important physical insights. It is particularly important that these programs be implemented in terminal rather than batch mode. The immediate interaction of the student and the program is essential. In order to save on terminal use and to help the students, we allow them to work in pairs on the computer exercises, although each must present his own write-up of the exercise.

When the students become familiar with the physics of a problem we proceed to discuss the computer techniques employed in solving that problem. Then the students are directed to write simple programs, or to modify the main program. Students have commented that it is at this stage that they discover what they really did or did not understand about the physics of the problem.

The computer programs in this manual are intended to have a dual role, teaching physics as well as computer techniques. We have thus attempted to use the most straightforward numerical techniques. We sacrifice efficiency for clarity in many cases. Thus, we use a modified Euler technique to solve differential equations while noting the existence of more complicated techniques which can be used where demanded by considerations of accuracy and efficiency.

An important part of the exercises in the student manual is the determination of the accuracy and the region of validity of the numerical techniques used. The appreciation, by the student, of the fallibility of numerical calculations, in certain regions, is a valuable lesson; besides which, the students derive a fiendish pleasure in discovering regions in which a program fails to work.

Students are occasionally asked to write or modify programs. At the most basic level they are asked to explain in words the operation of a program, or to convert from the dimensionless units used in most programs to S.I. units. At a more advanced level, students are asked to modify programs to work in a different coordinate system, or to write new programs. As an example, students were given a program which integrated in spherical coordinates to find the mass of a sphere given the density distribution. They were asked to modify the program to calculate the central moment of the sphere. Computer programming was not a prerequisite for the course. Most of the students learned to use BASIC rapidly. The computer language was not a barrier to their use of the computer.

### Transportability

The programs in this manual were written using a low-level subset of Dartmouth Basic which has proved to be easily transportable. We have already implemented these programs on Burroughs and Honeywell computers, on a Data General minicomputer, and on a Tectronix 4051 microcomputer. We highly recommend that students have access to a terminal to run these programs as compared to batch mode. The entire set of programs (without documentation statements) will fit into sixteen kilobytes of memory (sixteen bit bytes). Individual programs are much shorter. It should thus be possible to run these programs on the smallest minicomputer which can handle BASIC. The programs in this manual are written for teletype input-output. However, teletypes do not produce high quality graphics which can enhance the usefulness of many of these programs. Since there is no universal graphics language, we have chosen to separate the teletype graphics output from each program. This allows local graphics options to be added easily. At Oakland University we have implemented these programs on a Tectronix 4051 graphics system.

### Evaluation

The programs in this manual were developed and tested in small, special problems sections. They were then presented in two, junior-level physics courses in the fall terms, 1975 and 1976. These courses were based on a four-hour module including two, one-hour lectures and one, two-hour lecture recitation. The computer exercises were discussed during the two-hour session. Students averaged two hours per week of terminal time. They evaluate all of our physics courses at Oakland University. The two courses using this manual were both ranked two standard deviations above average in overall quality. Students commented that they were hesitant about the use of the computer at first. However, by the end of the term every student said that the computer modules should be kept as a major component of this course. Follow up interviews with students showed that every student who had not taken a course in computer programming before studying geophysics, signed up for a programming course the next term.

### Computer Animated Films, the Next Step

Some geophysics phenomena are so complex that even high quality computer graphics are not sufficient. What is needed is a rapidly varying computer graphics display. For example, if we are studying time independent heat flow in two-dimensions, one computer drawing of the isotherms would be sufficient. On the other hand, studying the time dependent heat flow from an igneous intrusion in two-dimensions requires a time varying graphics display, a computer animated "movie." Usually it is impossible to allow each student to generate such a movie in real time, since the time required to calculate each new frame is unacceptably long to the student at the terminal, not to mention the depletion of the physics department budget. Our solution to this problem is to produce a computer animated movie.

The main problem in making such a film is the low light level of most computer graphics terminals. Computer animated films have been produced for years in 16 mm format with very expensive equipment (variable-exposure-time single frame Bolex at $1,200). Super-8 copies are often produced from these originals. Recently, super-8 cameras with the necessary low light 120° shutter, f 1.2 fd st lens, macro-focusing, and single frame exposure, have become available for about two hundred dollars. We are producing computer-animated film loops using a Tectronix 4051 microcomputer and a Chinon 555X1 super-8 camera. The computer draws each frame, this process requires two minutes for the time dependent heat flow calculation, then it triggers the camera to take a single frame.[1] The time-dependent heat flow movie required eight hours to complete. This was accomplished in one night without supervision. The results are a qualified success. The resolution of the film, and the lens at f 1.2, is comparable to the resolution of the computer graphics display, this resolution is not sufficient to withstand projection on a large movie screen. However, the use of a super-8 film loop projector on a 26 inch diagonal screen produces clear movies. We thus plan to continue to make film loops for use in small recitation classes.

## The Manual

Attached is the Table of Contents for the student manual.

## Note

1. The printed circuit design for a trigger compatible with the IEEE 488 general-purpose interface bus is available upon request from the author.

## Appendix

## Student Manual: Table of Contents

### Foreword

1. Introduction
   - Units, dimensionless, and otherwise
   - numerical integration
   - numerical differentiation
   - example, the hyposmetric diagram
   - exercises
2. Gravity and the Figure of the Earth
   - The force of gravity, potential energy and equipotentials
   - The figure of the earth, flat earth, spherical and rotating self-gravitating sphere (fluid), tides
   - exercises
3. Gravity Anomalies and Isostasy
   - The Geoid, gravity corrections and anomalies
   - Anomalies of point masses
   - Anomalies of solid bodies, discs, finite cylinders, faults
   - exercises, mineral exploration, mascons
   - Isostacy, anomalies of bodies in isostatic equilibrium erosion of isostatic bodies
4. Magnetism
   - Magnetic field lines of dipoles and quadrupoles
   - The interaction of matter and the magnetic field
   - magnetic anomalies
   - exercises
5. Seismic Wave Propagation
   - P and S wave velocity and the properties of matter
   - Fermat's principle of least time, refraction at an interface, refraction in bulk media, reflection, the Moho
   - exercises
6. Seismology and the Earth's Interior
   - Seismic wave propagation in the real earth
   - The Williamson-Adams equation
   - The moment of inertia of complex bodies, the earth
   - exercises
7. Equilibrium Heat Flow
   - Radioactive heat sources, heat flow, and temperatures in slabs and spheres
   - Temperature versus depth in the earth
8. Time Dependent Heat Flow
   - Heat pulse due to igneous intrusion
   - Decaying heat sources, radioactive heating of the earth
   - The heat diffusion equation
   - Periodic heat source at a plane surface, Moon and Mars

# SPACETIME—An Experimental Learner-Controlled Dialog

**Alfred Bork**
*Physics Computer Development Project*
*University of California*
*Irvine, California 92717*
*(714) 833-6911*

## Modes of Computer Use in Learning

The traditional older pattern for student-computer dialogs, often called "CAI" or computer assisted instruction, was initially derived from programmed instruction. Although such material has been criticized, it is possible, even in this traditional pattern, to be much more versatile and responsive than in printed programmed instruction, responding in much greater detail to the needs of the student. Existing materials do not often show this great flexibility, however. Further, the traditional technique for computer use pushes students pretty much in the same paths, although alternate branches and even whole alternate segments of programs are possible and sometimes occur.

An alternate approach is sometimes described by the term "generative CAI." Here the program has a certain amount of intelligence in responding to student questions. It may have built in a full artificial intelligence facility for natural language recognition, and may have some structured knowledge of some discipline. Perhaps the best program of this type is Sophie developed by John Seeley Brown at Bolt, Beranek and Newman. The programs are often large and demanding, making them impractical today with large numbers of students. At the simplest level, generative CAI involves problem generators, giving the student a large collection of examples until a sufficient performance level is reached.

Other types of experiments giving the learner more control, but not as much, at least in principle, as in the generative environment, have also taken place. The TICCIT system, with its key pad of special buttons, even builds one such strategy directly into the hardware of the terminal; so all of its material follows a learner-controlled approach. Other experimentation in greatly increased learner control has been taking place here at the University of California, Irvine, in the development of the "new worlds" dialogs, the work of my colleague in the physics computer development project, Richard Ballard. These programs allow the student to move around very freely, changing the topic or obtaining assistance at any point. They are based on a two-level structure, the material the student is currently working at and a higher level structure that always gains control if the low level structure can't understand the response.[1] The higher level structure knows the vocabulary of various parts of the program.

These various strategies are often presented as if they were competing alternatives. Rather what appears to be the case when one actually examines the materials produced is that each has advantages for certain types of teaching situations, and that the materials produced by these different strategies are *by no means* equivalent pedagogically. Thus, the supporters of generative CAI often complain about the traditional frame-oriented CAI, particularly in terms of the amount of time it takes to write such material. Nevertheless, the material they typically produce undertakes different *learning tasks* than do most of the existing CAI materials.

Hence, the author would like to view the situation as not one of trying to find which of these methods is "best" but rather as one of finding which is most appropriate to specified learning situations. We have little experience so far, but progress is being made.

### Philosophies of Learning

What is sometimes lacking in these discussions of alternate computer strategies, particularly in the approaches from an artificial intelligence viewpoint, is some understanding of the learning situation of the student. Here we are in great difficulty because the process of learning is not a completely understood process, and different teachers, each with extremely good reputations, go about structuring learning in entirely different ways. Thus, teachers who are strong believers in the Socratic method in dealing with their own students—the constant questioning and asking of new questions of students as they talk to them in the office—will feel relatively at home with greatly enhanced and flexible program instruction capabilities. An instructor who feels that drill and practice are essential for students will probably find that much computer-based drill and practice material removes the tedious activities of the course from the instructor and gives them to the computer. Philosophies of learning which stress the dominance of the individual, his or her complete control, will feel happy with situations where the student can always change the subject with no difficulty.

Some patterns will probably work better in some pedagogical situations, some in others. Thus, if one is trying to teach problem-solving techniques to students, the kinds of strategies which may be most useful may be quite different from those which are trying to build up intuitive understanding of the phenomena. Both of these are extremely important goals in science courses at all levels, and they are also goals of great interest to our project at Irvine.[2]

One could, of course, approach the problem from an entirely different point of view, looking at some of the current attitudes toward *learning*, and using those to structure computer dialogs, either in the above forms or in the new forms. Bunderson's work takes that approach. The computer dialog described in this paper is another outcome of such a process, one which also involved some consideration of where the above strategies have their advantages and disadvantages.

### Piaget

One direction of current educational thinking that has had relatively little influence on the construction of student-computer dialogs are ideas of Jean Piaget, either his own or those of contemporary students of Piaget, particularly those involved in the development of curriculum materials. One project which has had very heavy Piaget influence, although not working with computers, is the Science Curriculum Improvement Project at the University of California, Berkeley, directed by Robert Karplus.

One aspect of Karplus's view of Piagetian ideas is to structure the learning process into three phases. The first phase might be described as the free play phase, where the student is gathering experiences and has complete control as to the nature of those experiences. In children these experiences are from the "vocal world," but they can also be provided via computer. In the past materials produced at Irvine our most heavily used dialog, MOTION,[3] is a good example of that approach. The students can explore the Newtonian "$F = ma$" universe in extremely easy to use and convenient fashion, changing the force laws, the constants in the force laws, or the initial conditions, and the variables plotted. Thus, they can start examining an inverse square force, can try different initial conditions, can ask what happens if the force is not an inverse square force, can look at the behavior of kinetic and potential energies as functions of time, can move to a damped driven oscillator, can look at it in phase space, first for the standard linear damping and then for nonlinear forms of damping.

The second aspect of Karplus's interpretation of the Piaget approach to learning is the formal exposition stage, where the student, presumably profiting from this free developmental play, gaining insight, experience, and motivation, then goes into much more formal learning sequences, which expound particular concepts, ideas, and logical developments. In this stage a variety of strategies can be followed. One, for example, is that the student should develop the *notion* of a concept before seeing the terminology. Or we can follow a more conventional route. Often mathematical derivations are made here, and the computer has some real advantages in allowing students to attempt the derivations themselves as much as possible, rather than simply passively seeing them in the text or the lecture. The type of learning—concept formation, logical structure, etc.—is that which textbooks and lecturers in the sciences are often most oriented toward. The tasks involved often are not easy, and in traditional modes of education some motivation is needed (even in some cases threats, primarily in the form of grades) to keep the student at such relatively difficult tasks. It seems to me unclear as to whether many students would ever learn difficult materials of this kind in a completely free environment, given the difficulties encountered.

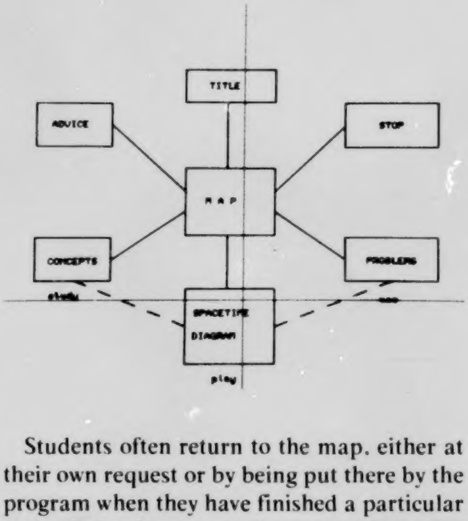
A third stage in this approach to Piaget's ideas is the *use* of the material. In science we are not interested simply in developing factual information or in developing definitions. We do not want memorizers. Our whole interest is in whether students can use what they already know to generate new situations that they have not already studied. This can occur at a very simple level, with the solutions of problems which involve only slight extensions of what the student has already seen in exposition; or this stage can become, in later life, a major research problem for a scientist. It is critical in science education to understand that we are not archivists, not simply interested in compiling "facts," but in developing new knowledge based on the use of older theories. Even beginning students are expected to be able to do this in limited and structured ways, and the goal is important at all levels.

## Strategy of Spacetime

With this discussion in mind we now describe the dialog, under development at Irvine, that tries to incorporate the Piaget pedagogical ideas just presented.

The subject area of this dialog is special relativity. In our curriculum at Irvine this material occupies five weeks in the second quarter of the five-quarter physics sequence for science and engineering majors. The text most commonly used at Irvine has been Taylor-Wheeler, "Spacetime Physics,"[4] and the dialog references this work.

The central part of the dialog is a map, as follows:



Students often return to the map, either at their own request or by being put there by the program when they have finished a particular task. Thus, the map is the central jumping-off point for the parts of the program. The three aspects at the bottom of the map indicate the three modes of learning just discussed. Students point to what they want with the crosshairs.

In this program the free play sequence is a space-time diagram, where the student is free to manipulate the diagram at will. The facilities involved will be described in a later section.

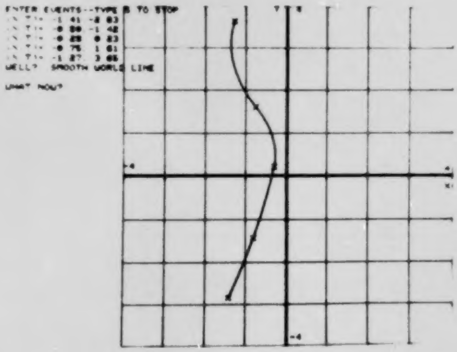
The program also has a series of concepts. Students can pick these by name; the program has enough intelligence to know which have already been studied, and to make decisions based on that. The student is not allowed to exit quickly from a concept. If there is a request to go back to the map in the middle of a concept, and then change the topic, we try to persuade the person to stay within the concept, sending the student back to the map only reluctantly. Strategies followed in the concepts themselves follow a wide variety of details.

The third phase is the problems, the applications of the material already learned. These problems are worked interactively at the terminal, with the computer offering guidance if difficulties arise. Some of them make use of the space-time diagram, but in a different way from the free play; the program always checks to see just what the student is doing with the space-time diagram, and offers advice based on these actions. Other types of problems make use of problem generators, providing drill and practice with some of the key notions of relativity, trying to analyze the types of difficulties the student is having in connection with problems of this type if the student has difficulties.

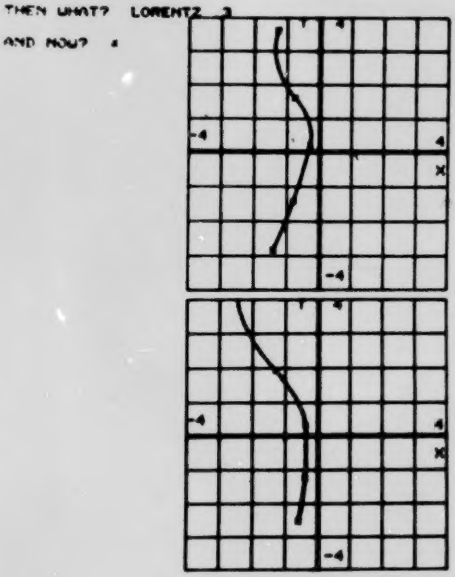
## Spacetime Diagram

The spacetime diagram used in this program, both in the free play section and in connection with the problems, is based on an earlier development by Edwin Taylor and colleagues at the Education Research Center at MIT. Work by Tim Kelley at Southern Oregon College also provided useful insights. Differences here are reflected both in computer hardware and pedagogical issues. The MIT project was done on a refreshed display, driven by a small, dedicated computer. At Irvine we work in a timesharing environment, using a Sigma 7, with over thirty graphic displays (Tektronix 4013s). The Tektronix display has a low information rate—1,200 baud—in its normal operation, and also cannot display information dynamically; the bistable storage tube stores the picture within the phosphor, so a true dynamic display is not possible. Our advantage is that we can provide access for more students than could be accommodated at MIT.

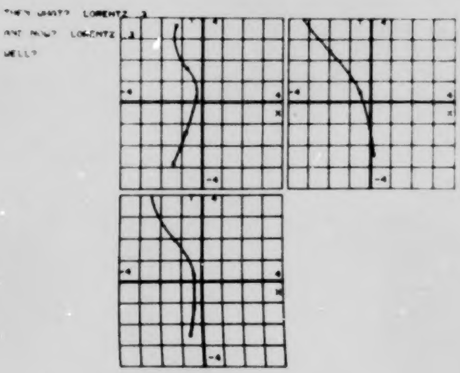
The space-time diagram facility lets the student enter events (both numerically and by pointing to them with the built-in crosshairs), delete events, construct world lines (smooth or straight line segments), enter invariant hyperbolas, and perform Lorentz transformations on whatever data are available, events, and world lines. Many events and world lines can be active at the same time. For beginners, a detailed set of directions is available on request, and the student can also request a briefer list of commands. Here are some example displays.



```
ENTER EVENTS--TYPE 0 TO STOP
X, T?: -1.41 -2.03
X, T?: -0.80 -1.42
X, T?: -0.88  0.83
X, T?: -0.75  1.61
X, T?: -1.87  3.06
WELL?  SMOOTH WORLD LINE

WHAT NOW?
```



```
THEN WHAT?  LORENTZ .3
AND NOW?  .
```



**Why is interval an important concept?**
**More generally, the student often wonders**
**why certain concepts are defined**
**and not others.**

**We now come to a key idea of relativity,**
**perhaps the key idea.**

**We introduce this new idea via a "thought"**
**experiment, as described in Taylor-Wheeler**
**SPACETIME PHYSICS.**

**PRESS RETURN TO CONTINUE...**

### Concepts

The formal exposition of the ideas of relativity is contained in the concept section of the program. The following concepts are either available or contemplated in development of the program.

**CONCEPTS**

**CENTRAL IDEAS OF SPECIAL RELATIVITY**

1. **Events.**
2. **Frames of Reference.**
3. **Interval.**
4. **Invariance of Interval.**
5. **Lorentz Transformations.**
6. **Length contraction.**
7. **Lorentz Transformations Revisited. (Not available)**
8. **Vectors. (Not available)**
9. **Tensors. (Not available)**
10. **Mass, Momentum, and Energy. (Not available)**
11. **Form Invariance Under Lorentz Transformations.**
12. **Relativistic Theories. (Not available)**

**Which one?**

The student can request any concept, although naturally some may depend conceptually on earlier concepts. It may be suggested to students in trouble that they go back and pick up earlier concepts. Typically a concept sequence is not long—perhaps fifteen minutes at the terminal, although with certain critical concepts, such as Lorentz transformations, the time will be greater. The student can exit from a concept, but only after being explicitly advised that we don't think it's a reasonable idea to take that course of action. Some of the concepts are developed in the form of experiments. Thus, the study of invariance of interval is based on an experiment in a rocket ship described in Taylor-Wheeler.[4]

Eventually we hope to include many more concepts and problems; the program can grow without limit.

### Problems

About seven problems have already been written, and others are under consideration. Some problems make use of the space-time diagram as described. However, when students in this case use the space-time diagram, the computer program "watches" what is happening, and takes appropriate action based on the sequence on student inputs. Thus, when the user is asked to enter events at certain positions and times, the computer checks to see whether the correct event is entered and if the student really knows how to use the space-time diagram. In some of the problems the space-time diagram is used empirically to let students gain experience in results that they will later be able to obtain analytically.

Other problems are based around problem generators. The student is asked in one problem to identify a linear transformation as being Lorentz or not. A variety of cases are given, and students must perform correctly on a number of them before the program decides that they understand the concept and sends the student to some other area of the program. For other students, the program offers some assistance in how to recognize Lorentz transformations.

In dealing with the problems the program examines which of the concepts the student has studied, and may take alternate actions or use alternate terminology depending on the performance of the student on these concepts. It is planned that the program will automatically have long-range memory, so that when the student comes in again, the computer will be fully equipped with knowledge of past performance. This facility is currently under development.

### Advice

As indicated on the map, one section of the program attempts to advise students as to what they are doing. Again, this advice is based on the internal knowledge of the program as to what the student has already done. This section also expounds some of the underlying educational philosophy of the program, as explained here.

This is a part of the program which in some ways is the hardest to write initially. The author has a difficult time speculating as to just what kind of advice students are seeking. Hence, this section will be particularly interested in querying the student, and storing replies, so that the later versions of the program can be more responsive to the type of advice and help directly needed by students who use the program. This will be done extensively in all parts of the program; such feedback has proven extremely valuable in improving computer-student dialogs.

### Future Developments

As indicated, SPACE is still in a relatively early stage of development. We expect to continue adding material to it, both new concepts and new problems. Because of the overlay structure of the program, its length could be increased without end, so that it might eventually go to advanced concepts of relativity and to more difficult problems.

### Acknowledgments

Estelle Warner is the principal programmer in the Physics Computer Development Project. Stefan Demetrescu is responsible for this particular dialog. The project is supported by the National Science Foundation (Grant No. EC44061) and the University of California.

### References

1. Ballard, R. L., "Authoring New Worlds," March 3, 1975.
2. Bork, A., "Physics Computer Development Project," Computer Graphics and Art for August, 1976.
3. Bork, A., "Effective Computer Use in Physics Education," Am. J. Phys., 43, No. 1, January 1975.
4. Taylor, E. F. and Wheeler, J. A., *Spacetime Physics* (W. H. Freeman and Company, San Francisco), 1963.

# Course Management System for Physics III at Irvine

**Alfred Bork**
**Jeff East**
**Joseph Marasco**[1]
*Physics Computer Development Project*
*University of California, Irvine*
*Irvine, California 92717*
*(714) 833-6911*

## The Course

Physics III is a large (approximately 300 students) course taken primarily by junior and senior biological science majors. The course is a three quarter, calculus-based introduction to physics. A complete description of the course may be found elsewhere.[2]

Two things distinguish this course from most others. For the fall (first) quarter (which deals with mechanics) the course is self-paced and multi-tracked. This means that students have some control of the speed with which they move through the material (self-pacing) and in addition, some control of the content. Not all students see the same material, and different students may see different topics at different times. Figure 1 shows the various allowed paths through the course, breaking it down into blocks called units.

Clearly the variety allowed students increases the complexity of record keeping. Students indicate progress by passing quizzes (typically one or two per unit). By the middle of the course students are in as many as ten different units. Furthermore, since students often have to take a given quiz several times before passing it, multiple versions of written quizzes are required, as well as records of which version of the quiz the student has already seen.

We decided after an initial offering of the course that traditional methods of administering, grading, and keeping track of the quizzes were grossly insufficient and that much could be gained by going to a system where all the quizzes were given on-line. This has several advantages, the most important being the possibility of offering every student a unique quiz on each attempt. For pedagogic as well as practical reasons, we wanted to control access to the quizzes; for example, we wanted students to wait a day before retrying a quiz which was failed. This paper describes the computer-based system designed and implemented to handle the problem.

## The System—An Overview

We will first attempt to describe the system in general, showing the relation of the various components. We will then talk in more detail about some of the particular aspects of the interfacing with existing software.

Figure 2 shows the structure of the course management system. the principal file is the Extended Data Management System (EDMS) data base, a commercial data management system provided with the Sigma 7 employed by this project. (This file may be found near the center of the figure.) The data base contains all the student records for the course; a complete description of its network structure is beyond the scope of this paper but is available on request from the authors. One of our guiding design principles was that although the data base could be examined at any time, we wished to update it only once a day. This criterion grew out of cost and security considerations.

How does EDMS get updated? Working backwards on Figure 2, we see that a batch job called the DATABASE UPDATER reads a file called the DAYFILE and then updates the data base. (This and several affiliated jobs are run by our computer center as part of the end-of-day procedures.) As indicated in the figure, any errors or anomalies encountered during the update are written out to a file; the old DAYFILE and the entire new data base are copied to tape as a backup. In addition, a fresh (empty) DAYFILE is created for the next day, as well as something called an authorization file. This entire process is loosely called "updating the data base," and it occurs every twenty-four hours.

Crucial to the above process is the notion of the DAYFILE. Two types of user programs write on the DAYFILE. First, all the on-line quizzes write their results onto it. Second, the instructor, using the program DATLOAD, can enter off-line results and make corrections. Note that both the quizzes and DATLOAD are interactive, on-line programs that are available at all times. (DATLOAD is a restricted access progrm that only the instructor and other authorized persons may use. A student's complete record may be accessed by DATLOAD; Figure 3 is a typical example.) Material entered by both these methods shows up in the data base the next day, after the data base has been updated.

How does all this appear to students? Their perception of the system is through two channels—the first being the quizzes themselves. They are allowed to take quizzes based on previous performance: quizzes become authorized progressively. Hence there is a quiz-authorization file (see Figure 2 again) which permits or interdicts student entry to a given quiz. Also, passing a quiz may open other quizzes, so the quzzes write on as well as read from the quiz authorization file. This file is recreated each time the data base is updated; an on-line authorization file maintenance program allows the instructor to remedy problems needing faster than twenty-four hour turnaround.

The second program accessible to students is P3A, which allows them to examine their own records from the data base. Figure 4 gives an example of what the student sees. Any discrepancies noted by the student are written out to a file called PROBLEM which may be accessed by the instructor, who can then make corrections via DATLOAD.

The only remaining part of Figure 2 yet unexplained concerns the gathering of statistics. A batch job called PSISTAT reads from the data base and writes two summary files, one showing the status of each student and one reviewing each unit and quiz. This batch job is run once a week. Hard copy of these statistics is available from the line printer. In addition, a graphics display program, PSIG, uses the information in the summary files to construct histograms and scatter plots for the course; Figures 5 and 6 show typical output. A map showing the progress of the students through the course can also be generated. Figure 7 shows the location of students near the end of the quarter. The numbers refer to the number of student at the particular point on the map.

### Some Further Details

To monitor student progress through the course and to gather meaningful statistics, a number of new macro instructions were implemented to supplement the collection already available as part of our DIALOG software. We will describe the three principal ones to show the relative ease with which we were able to add these features to our existing system.

The first new macro is called QUIZBEGIN. QUIZBEGIN performs several important functions. It marks the point at which an instructional segment shifts from being a conversation-like dialog—hence free accessible—to a quiz segment. QUIZBEGIN first searches the authorization file to see if the person's account matches an account listed there. If so, further checks are made to see if the student can take the quiz at that time. If the account from whence the QUIZBEGIN query originated is not listed in the authorization file, a search is then made in a privileged user file. Listed here are the accounts of instructional personnel who need to be able to access any quiz at any time for reference or demonstration purposes. Should the account number not be found either in the authorization file or in the privilege file, access is denied and a message is sent to the user.

QUIZBEGIN also sets the time limit allowed for the quiz, does some bookkeeping with the authorization file to indicate the quiz has been attempted and sets a flag indicating whether the user was authorized or privileged. No further records are kept for privileged users. Note (Figure 8) that QUIZBEGIN talks only to the authorization file and not to the DAYFILE.

A second macro, QUIZMAP, aids in keeping detailed records of student progress through the quizzes. Each quiz has this instruction embedded at several crucial branching points. By writing a unique identifier for each occurrence, we can save the student's path through the exam, consisting of what problems have been attempted, what problems have been successfully completed, how many attempts have been made on each segment, etc. Clearly this is a large amount of data—a "map" of a single student's progress through a single quiz can consist of ten or more calls to QUIZMAP. Considering that we administered over 10,000 quizzes in one quarter, the question arises as to where to store all this data. We choose not to keep this in the data base but to use our nightly archival tape save of the DAYFILE to permanently store this data. Future users of this data need only to read the DAYFILE tapes to extract it.

The third important new macro is QUIZEND, our closing procedure on quizzes. Under normal circumstances, each quiz ends with a call to QUIZEND. Depending on arguments set by the quiz, QUIZEND records the result in the DAYFILE: Pass, No Pass, or No Pass/Time Limit Exceeded. Modifications, when necessary, are made in the authorization file. A message indicating the action taken is sent to the student. In the case of an abortive exit—the student breaks out of the program or shuts off the terminal—our software allows us to get control and execute the appropriate QUIZEND. Note that in the event of a system crash, nothing gets written in the DAYFILE—there is no record of the interaction. To allow the student access to the same quiz again—since a QUIZBEGIN removing authorization for further tries *has* been executed—we adopted the following procedure. Every hour we automatically run a job recopying the current authorization file. In the event of a crash, the computer center operators are instructed to run a special job on restart which reloads the last copied authorization file. Hence, quizzes interrupted by the crash can be immediately retaken once the system comes up again.

## The Next Iteration

The system described thus far worked quite well during the fall of 1976. Over 10,000 on-line quizzes and several hundred off-line exercises were entered, as well as scores for optional units and the final exam. More than 300 students were enrolled initially in the course and about 275 completed it. In spite of the usual number of crashes and system problems (including some rather acute problems with disks) the data base survived intact.

Nonetheless, there are some things we discovered which we are implementing for the next go-around which may be of interest to the potential course management system builder. While most of these are particular to our own way of doing things, they may in fact be generic in that all newcomers may tend to commit the same oversights we did.

One of the things that became painfully obvious as the quarter wore on was that our method of handling the diagnostic messages from the updates was poor. That file was overwritten each night, making it necessary for someone to examine it every morning before it was overwritten the following night. A better system would be to append new messages onto the old ones, and to create a structure which allowed the person correcting the errors to simultaneously record the action he had taken and delete the diagnostic from the file. As in any large system where there are multiple principal users, it becomes imperative to have a good way of recording actions taken so as to ensure that all students are being taken care of and that things are being done in a uniform way.

Another flaw was the system of only allowing students to go on to the next unit after all material was completed in the present one. While this worked well for the on-line materials, problems arose for the off-line assignments. Students who previously had waited up to a week to have homeworks graded in other courses became impatient when an assignment handed in did not appear the next day in the data base, allowing them to proceed. This was no fault of the computer or the data base: the weak link was the real-time needed to grade and record the work. To combat this problem, our ever-efficient students started to hand in this work very early. Unfortunately, we were then caught in our own net for the opposite reason: the work belonged to the next unit and hence was rejected by the data base because it was too early! The moral of the story seems to be that off-line material is best handled by a system that is no more rigid than you really want it to be. On our next iteration we intend to have two parallel streams: one which has a strict time sequence and one which allows material to be handed in and entered at any time.

A final observation is probably in order as to some sociological effects. Students responded favorably to having access to their records at all times, but expectations were very definitely raised as to our ability to respond to any and all problems that arose. Since students had virtually twenty-four hour access to the quizzes, they sometimes expected that someone would always be there in case of problems. We were highly visible, accessible, and eager to iron out the minor difficulties which came up, yet in some instances students felt frustrated that we weren't doing more. The fact that more attention was being paid to them than in most traditionally-structured courses seemed to be little remembered. Caveat emptor!

## Notes

1. Correspondence regarding the contents of this paper should be directed to Professor Marasco.
2. "Modes of Computer Usage in Physics," A. Bork and J. Marasco, *The Journal*, February 1977.

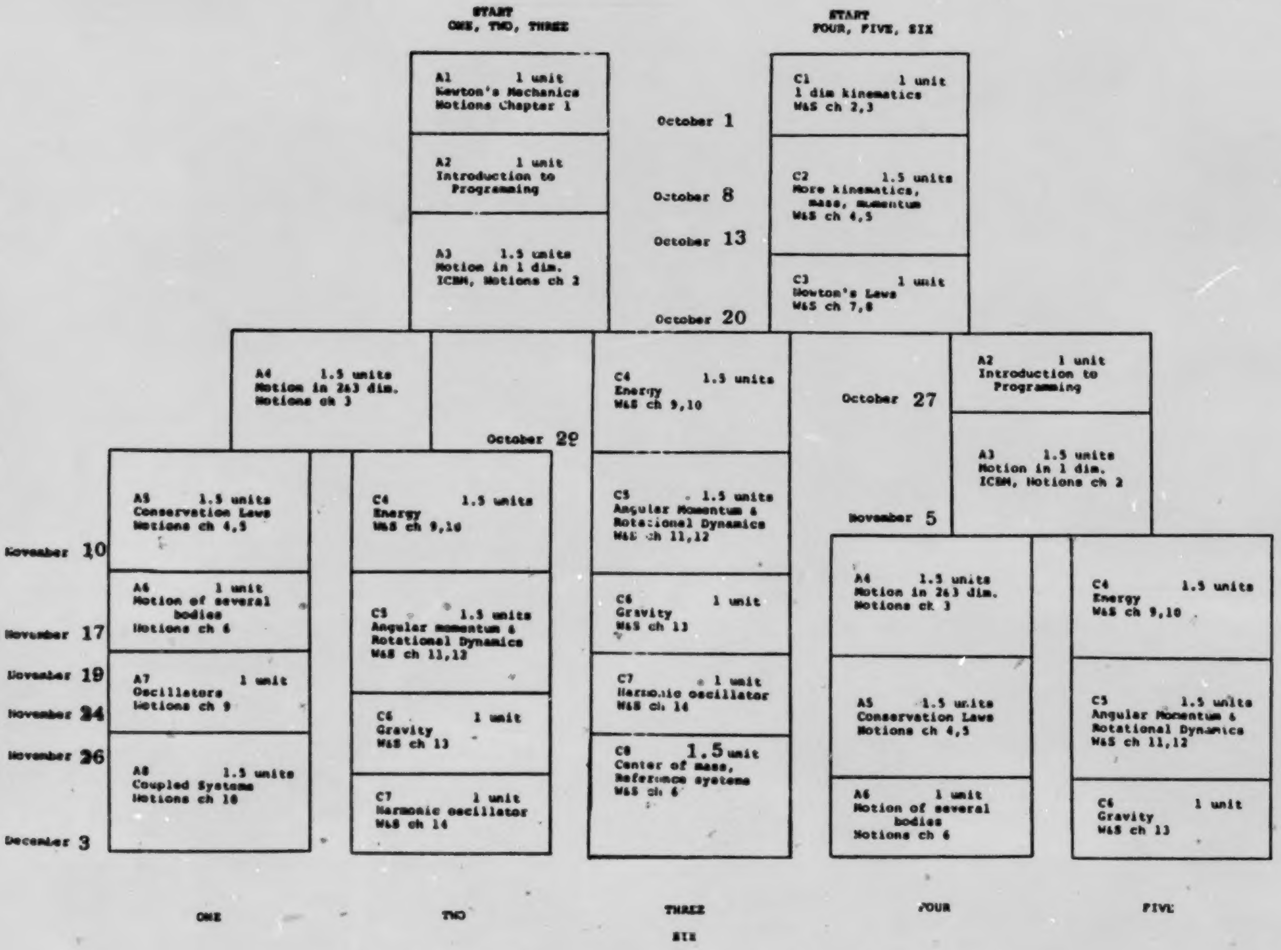


Figure 1. Structure of the Multi-tracked Self-paced Physics III Course.

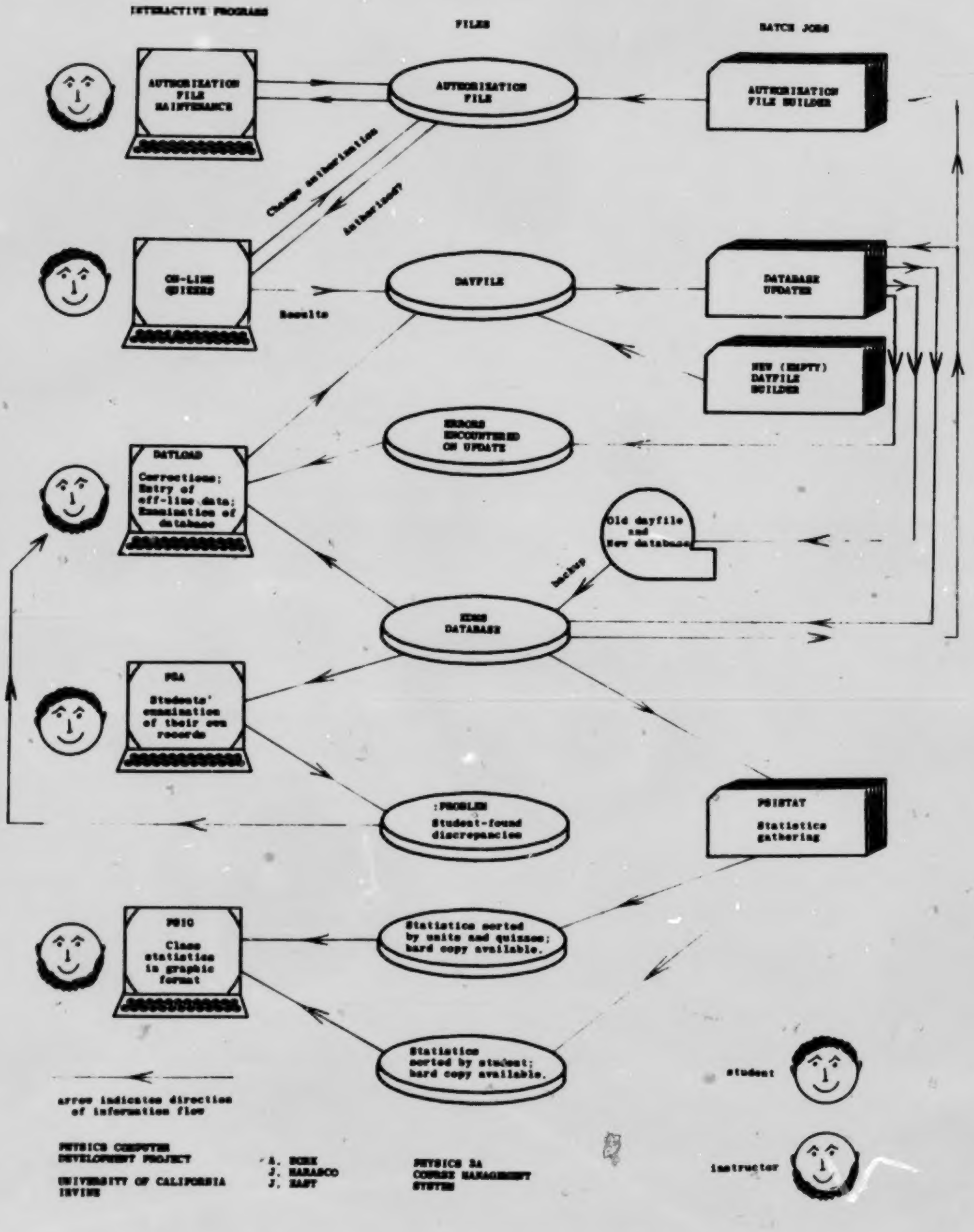
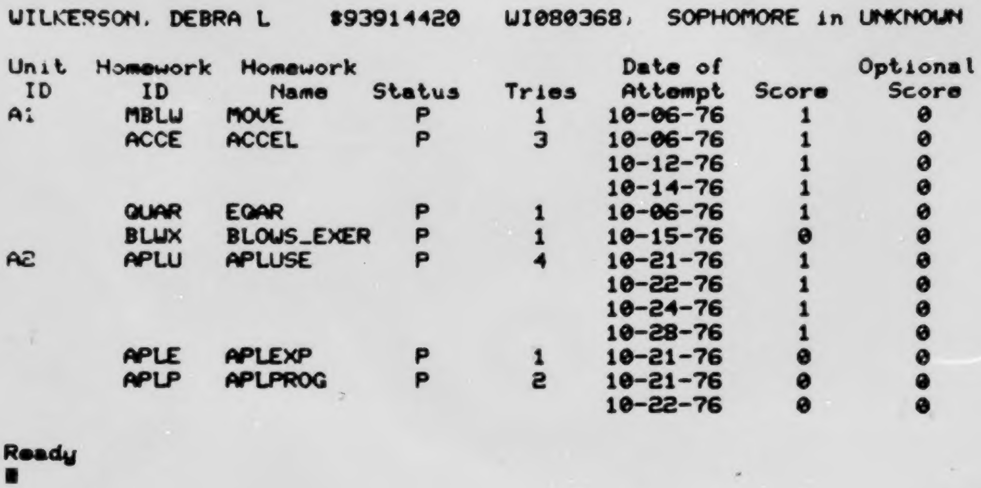
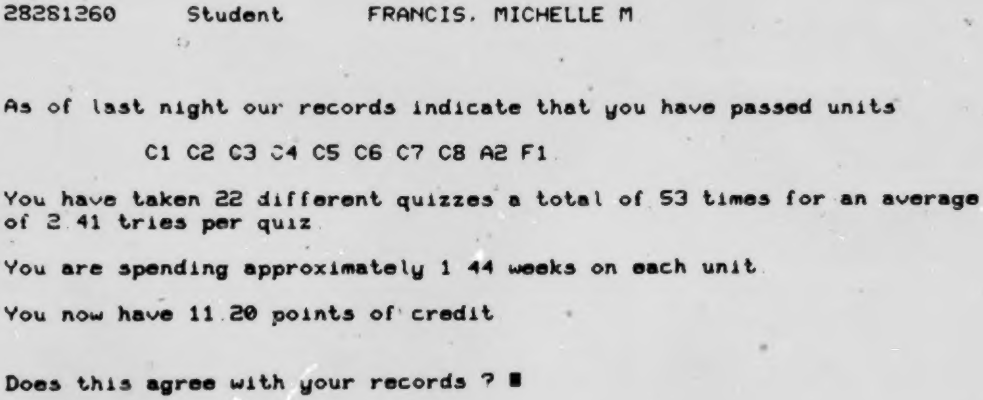

INTERACTIVE PROGRAMS
FILES
BATCH JOBS
AUTHORIZATION FILE MAINTENANCE
AUTHORIZATION FILE
AUTHORIZATION FILE BUILDER
Change authorization
Authorized?
ON-LINE QUIZZES
Results
DAYFILE
DATABASE UPDATER
NEW (EMPTY) DAYFILE BUILDER
ERRORS ENCOUNTERED ON UPDATE
DATLOAD
Corrections; Entry of off-line data; Examination of database
Old dayfile and New database
backup
IDMS DATABASE
PSA
Students' examination of their own records
:PROBLEM
Student-found discrepancies
PSISTAT
Statistics gathering
PSIG
Class statistics in graphic format
Statistics sorted by units and quizzes; hard copy available.
Statistics sorted by student; hard copy available.
arrow indicates direction of information flow
student
instructor
PHYSICS COMPUTER DEVELOPMENT PROJECT
UNIVERSITY OF CALIFORNIA IRVINE
A. BORK
J. MARASCO
J. EAST
PHYSICS 3A COURSE MANAGEMENT SYSTEM

Figure 2. Structure of the Course Management System



```
WILKERSON, DEBRA L    893914420   WI080368,  SOPHOMORE in UNKNOWN

Unit  Homework  Homework                          Date of           Optional
 ID      ID         Name    Status     Tries      Attempt    Score     Score
A1      MBLW    MOVE           P          1      10-06-76      1          0
        ACCE    ACCEL          P          3      10-06-76      1          0
                                                 10-12-76      1          0
                                                 10-14-76      1          0
        QUAR    EQAR           P          1      10-06-76      1          0
        BLWX    BLOWS_EXER     P          1      10-15-76      0          0
A2      APLU    APLUSE         P          4      10-21-76      1          0
                                                 10-22-76      1          0
                                                 10-24-76      1          0
                                                 10-28-76      1          0
        APLE    APLEXP         P          1      10-21-76      0          0
        APLP    APLPROG        P          2      10-21-76      0          0
                                                 10-22-76      0          0

Ready
■
```

Figure 3. A Typical Student's Record Examined in DATLOAD



```
28281260      Student      FRANCIS, MICHELLE M


As of last night our records indicate that you have passed units

          C1 C2 C3 C4 C5 C6 C7 C8 A2 F1

You have taken 22 different quizzes a total of 53 times for an average
of 2.41 tries per quiz

You are spending approximately 1.44 weeks on each unit

You now have 11.20 points of credit


Does this agree with your records ? ■
```

Figure 4. A Typical Response to a Student's Query to P3A

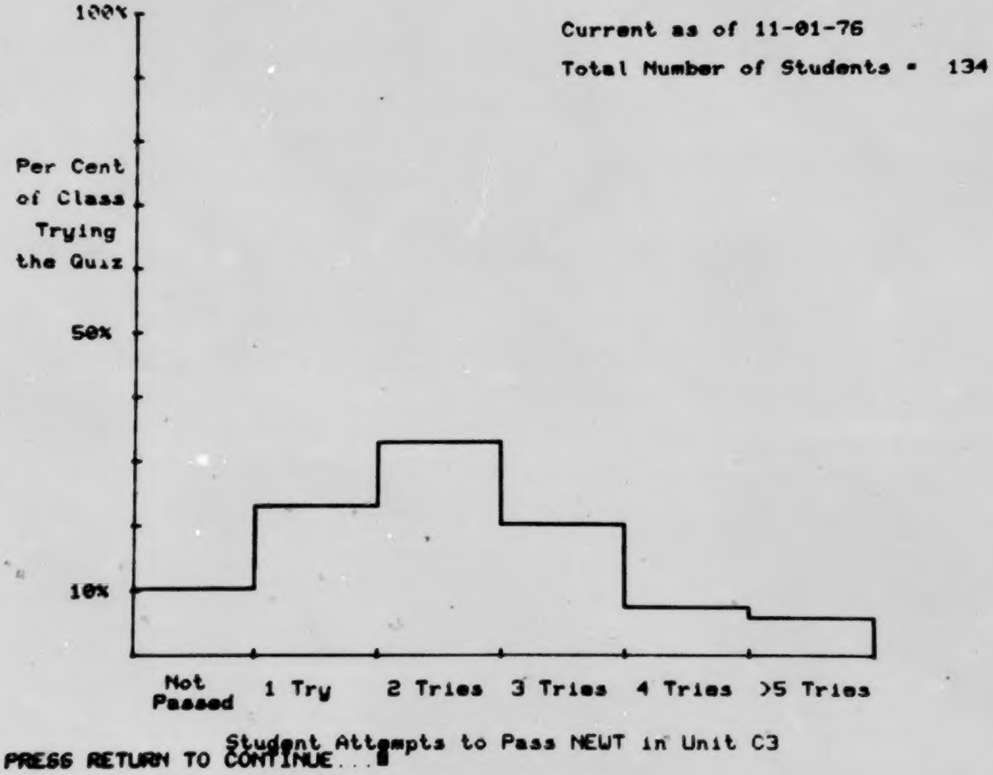
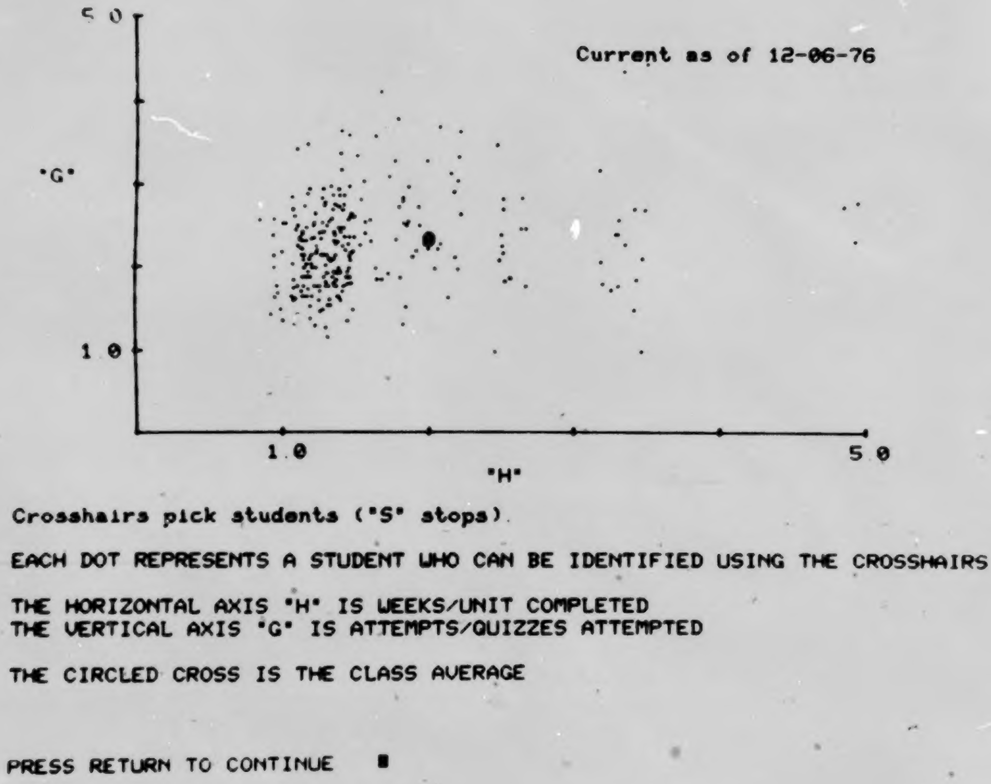
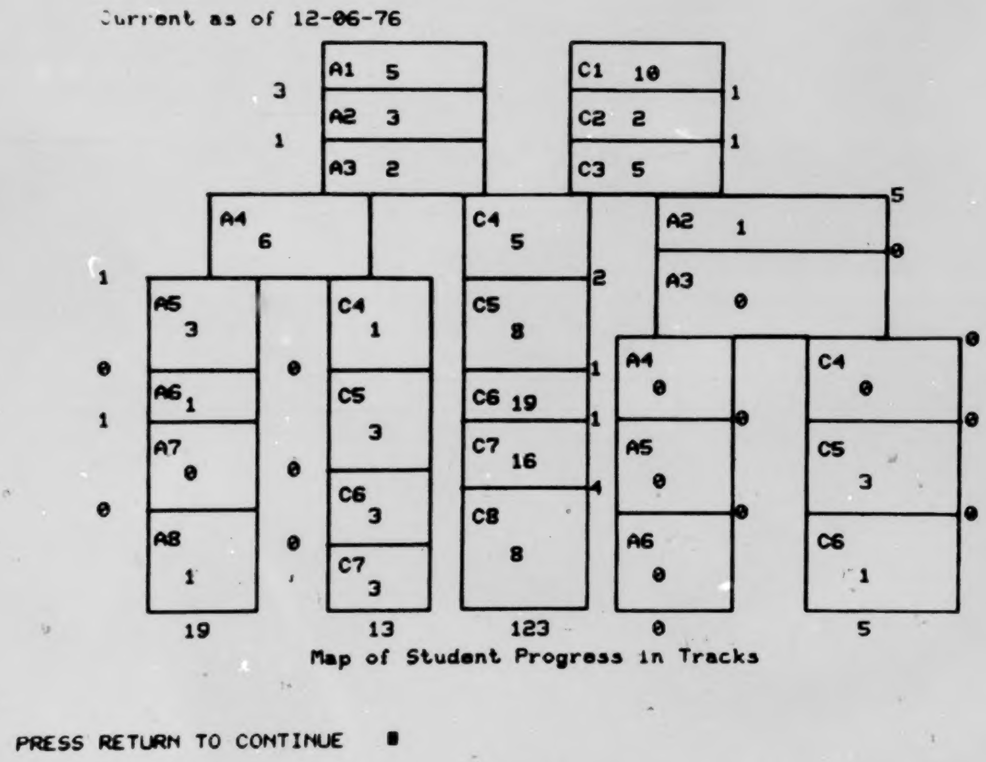


Figure 5. A Histogram for the Quiz NEWT Displayed by PSIG



```
Current as of 12-06-76

Crosshairs pick students ("S" stops).

EACH DOT REPRESENTS A STUDENT WHO CAN BE IDENTIFIED USING THE CROSSHAIRS

THE HORIZONTAL AXIS "H" IS WEEKS/UNIT COMPLETED
THE VERTICAL AXIS "G" IS ATTEMPTS/QUIZZES ATTEMPTED

THE CIRCLED CROSS IS THE CLASS AVERAGE


PRESS RETURN TO CONTINUE   ■
```

**Figure 6. Scatter Plot for the Entire Class, Displayed by PSIG**



Current as of 12-06-76

Map of Student Progress in Tracks

PRESS RETURN TO CONTINUE ■

Figure 7. Map of Student Progress Displayed by PSIG

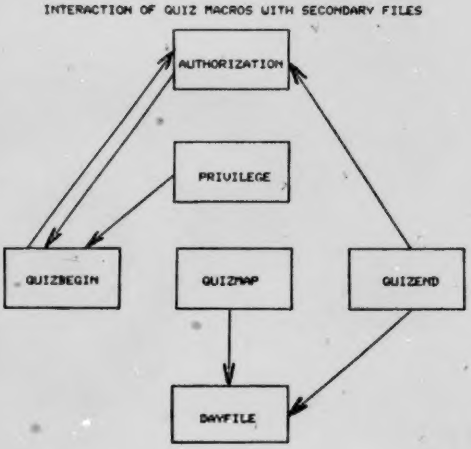


Figure 8. Relation of Quiz Macros to Secondary Files

# The Role of Computer Supported Instruction in a Course on the Psychology of Oneself

**Eileen G. Thompson**
*Computer Institute for Social Science Research*
*Michigan State University*
*East Lansing, Michigan 48824*
*(517) 353-2040*

## Introduction

While computer-supported instruction (CSI) has been the focus of growing attention in educational innovation, its use has tended to be restricted to specific types of teaching situations. CSI has been applied primarily in two areas. The first is programmed instruction, used to teach material in which concepts are specific and easily defined and for which fairly concrete behavioral objectives can be supplied. Programmed instruction has received wide interest in the teaching of mathematics, natural science, languages, etc. For psychology, the use of programmed instruction has been limited primarily to the teaching of introductory concepts in the "hard," i.e., physiological and learning, subfields of the discipline. The second application of CSI has focused on skill acquisition. Here the computer is used to help students gain familiarity with such complex skills as computer programming and statistical analysis. The skill acquisition applications of CSI have been used in psychology primarily in methods courses and have focused on experimental design and statistics.

Insofar as they do not fit either of these patterns, courses in the humanities and the "soft" social sciences have made little use of CSI. In these latter courses, particularly at the upper levels, an emphasis is placed on (1) the complex conceptual integration of substantial bodies of information, and (2) the process of applying concepts to the understanding of complex situations.

The purpose of this paper is to outline an approach to the use of CSI in a course designed to accomplish these kinds of objectives. The software which was developed for this application is called PROCTR and is described in a paper by Price (1977). The course for which the PROCTR application was initially developed is an upper-level class in personality and social psychology. The class is a two-term course, titled "The Psychology of Yourself" and offered under the Psychology 400H option, a special-topics honors seminar. The course has two primary purposes: (1) to help students understand, through dealing with real and self-relevant data, the interrelationships among a number of variables which are of empirical and theoretical importance in the areas of personality and social psychology; and (2) to help students apply this knowledge to achieve a deeper understanding of their own personalities and their relationships with others.

The framework of the course is quite simple. During the first ten-week term the students complete a large number of personality-assessment and social-psychological instruments. During the second term each student analyzes his or her own data and examines the interrelationships among the pieces of information obtained, interpreting the results in terms of self-knowledge and knowledge of personal events. The broad areas of psychological functioning which are included as topics in the course, each examined using multiple assessment measures, include: self-esteem, locus of control, social power, achievement orientation, personal values, sex-role identification, anxiety, sense of purpose, future time orientation, cognitive complexity/flexibility, and implicit theories of personality (personal constructs).

### Implementation

The data for the class are collected using a mini-computer with terminal interfaces. The breadth of the scope of this class is such that large quantities of data are clearly generated. Over the first term the students complete more than ninety tests or instruments, ranging from 2 to 600 items each. This term's work results in approximately 450 card-images of data per student. The collection and analysis of these data using ordinary procedures (data-coding and punching or OPSCAN materials) would be both expensive and cumbersome. Use of the mini-computer for the collection and primary analysis of these data bring the cost and data handling time within reason. In addition, the data may be transferred with a minimum of processing to the university's large computer, a CDC6500. This permits complex data analysis procedures—factor analysis, partial correlation, etc.—to be performed on the data, again at minimal cost.

The instruments are presented to the students and their responses recorded through the use of PROCTR, a BASIC program maintained on an HP2000 computer. The test measures are recorded on files, entered by use of an editing program. To take the instruments, the students use the facilities of the Behavioral Science Instructional Laboratory (BSIL), which consists of five CRT's and seven hard-copy terminals. After an initial in-class demonstration, the students complete the test instruments at times convenient to them. In order to complete the eight to ten instruments assigned each week, the students spend approximately two hours per week in the laboratory.

A student wishing to take an instrument on this system simply logs in on the computer using the class account and then types in the class ID and his or her individual ID number. The student then types in an identification for the test he or she wishes to take. Instructions for responding to the instrument are displayed or printed. The test questions are given, one at a time. When a response to a question has been recorded, the next question is presented.[1] Once an instrument has been completed, the student is asked if he or she wishes to take another instrument, and the process is repeated. For each instrument, responses are recorded on a *response file* along with student ID, instrument ID, date, time, and time required to complete the instrument.

The psychological tests presented vary widely in structure and format. The most commonly used instruments present questions or statements and require (1) a true-false response, (2) the use of a Likert scale (for example, "(a) agree strongly, (b) agree, (c) neither agree nor disagree, (d) disagree, (e) disagree strongly") or (3) an integer response indicating probability, certainty, or the percent of the time a test item applies. Another type of instrument which is heavily used is the semantic differential. In tests of this type, the student is presented with a concept followed by a series of response scales to be used to rate the concept. Each scale is anchored by two adjectives which are opposite in meaning (for example, "responsible /.1./.2./.3./.4./.5./.6./.7./ irresponsible"). Other instruments presented require the students to respond with words or sentences.

For example, a student is given the names of two acquaintances and is asked to supply a word expressing the relationship between them. PROCTR is designed to administer and record data from these various types of instruments.

#### *Special Features of the PROCTR Program*

The details of the PROCTR program are presented in Price (1977). However, several points concerning the efficiency and effectiveness of this mode of presentation of psychological tests should be mentioned here. A number of options are available in the program to facilitate data processing and to minimize the amount of "bad" or missing data. First, the use of the terminals to present the tests prevents the student from skipping questions, answering them out of order, or recording responses in an incorrect spot on an answer sheet. PROCTR includes a range check option, so that out-of-range responses and other types of response errors are not recorded—instead, an in-range response is requested. In addition, PROCTR includes a *reflect* option which substantially increases the ease of data analysis. This option reverses the response within the response range—if the response scale ranges from one to seven, a seven response is reflected to a one, a two response to a six, etc. The reflection facility is highly useful because many assessment instruments include both positively ("On the whole, I feel I am a worthwhile person.") and negatively ("I often feel I can't do anything right.") worded items. If negatively worded items are reflected, the recorded responses may simply be summed to obtain the test score.

Two other options available on PROCTR provide experimental controls which are not available using traditional means of test administration. The *random* option allows questions to be presented in random order, with responses being recorded in the original order. The *clear* option, available for use on CRT's, allows only a single question to be displayed on the screen at one time. Previous questions and responses are thus not available to the student.

For instruments in which the test score consists of simple summation of either the total set of responses or of various subscales of the test, PROCTR then completes the analysis. A *sum* option allows the instructor to specify one or more sums to be obtained by adding the responses from specified test questions. These sums are then recorded on the response file. This sums option is able to provide test score(s) for nearly every standardized psychological test.[2] For such standardized tests, then, the student's results may be obtained either immediately or at some later time, simply by listing the response file.

### *Instrument Structure*

A note should be added concerning the process of creating a file for a particular instrument. The instrument file, simply typed in using an editing program, contains two parts. The first section contains *specification* and *instruction* information. This section contains the instrument ID, response range information, choice of options such as randomization and sums, general instructions, and the response scale. Following this section, test items are entered, along with *reflect* and/or *clear* specifications for each question. The process of creating a new instrument file requires less time and no more specialized skills than typing stencils and mimeographing the test questions. Many of the instruments used in the class were entered by independent study students with minimum typing skills and no computing experience.

### *The Development of Individualized Instruments*

The procedures described above summarize the process of data collection and summary for standardized psychological instruments. Use of the computer, however, makes it possible to develop more individualized and complex psychological instruments. These kinds of psychological measures have been used extensively in the current course. An example of such a testing procedure is the Kelly Rep Test (Kelly, 1955). Here each student is asked to list characteristics for particular persons he or she knows and then to rate all the persons on each characteristics. In Psychology 400H, individualized instruments have been constructed, using each student's *own* list of concepts, for the rating of friends and acquaintances, aspects of the self, and future self-goals on semantic differential scales. In addition, each student has rated the degree of conflict among pairs of individually-listed hopes and fears and among pairs of current and future aspects of the self. A set of programs for the creation of these instruments is available. In each case, only the concepts for each student need to be entered, and it is possible for the student to create this type of an instrument himself or herself. The availability of the computer has made feasible this type of individualized testing, and the use of such techniques may be expected to increase.

## Analysis

As was mentioned above, for nearly all the standardized instruments PROCTR computes test scores through the SUMS option. For analysis of data from the semantic differential instruments and other individualized tests, the use of a large computing facility is required. These data are prepared for analysis on the CDC6500 computer through use of a program called MERGE. Through either remote job entry or the use of magnetic tape, the data are entered onto the 6500. They are then in a form which can be entered into a packaged statistical program such as SPSS. In the current course, heavy use is made of factor analysis of an individual student's semantic differential ratings. this procedure provides a representation of the constructs used by the student in rating a particular set of concepts—for example, the relationships among the traits used to characterize acquaintances. The individualized analysis of data from such individualized instruments would be difficult using traditional data collection and handling procedures.

The large computer may also be used to perform statistical procedures on the data from the entire class of students. Correlation and regression techniques may be used to examine relationships among the large number of measures for which data has been obtained in the course.

### Presenting the Data to the Students

The software used in the data collection, handling, and analysis are discussed more fully in Price, 1977. The author would like to touch briefly on the teaching of the class—what we do with the data, an outline of class procedures—and then recount briefly students' reactions to the course.

In the second term of the course, with the data available, the real work begins. This term the class meets for three to four hours per week, approximately one to one and one-half hours is spent in lecture, the rest in discussion. The lectures cover two topics: (1) the role of the particular concept under discussion—self-esteem, implicit personality theories, etc.—in the literature in psychology, and (2) the particular measures used, their interrelationships, and previous empirical results. The class discussion focuses on the self-relevant aspects of the data. In this discussion the students are encouraged both to examine aspects of themselves in light of the data and to examine critically the particular testing procedures in light of what they know about themselves. A final paper is required the second term. this paper is to be an "educated autobiography," describing the relationship between the psychological concepts and the student's own view of his or her personality and experiences.

The content-oriented class objectives of the course involve (1) knowledge of the role of the constructs being studied in current empirical and theoretical work in psychology, (2) knowledge of the multiple ways in which these constructs have been operationalized, and (3) understanding the effects of different operationalization procedures on the interrelationships among these constructs and on the relationships between the constructs and other psychological variables. For example, among a number of self-esteem measures, the overall level of intercorrelation is not high, but certain measures do form intercorrelated clusters. In the class lectures, we examine particular questions used in the various instruments, noting that some focus on overall sense of self-satisfaction, others on social self-confidence. These results are used to examine the issue of the unitary quality of self-esteem, and the implications of the results for theoretical work involving self-esteem are discussed.

The self-oriented objectives of the class include, for each student, (1) assessment of the interrelationships among test scores *in light* of the student's own knowledge of himself/herself, (2) comparison of variation in test scores over time with the variation in experiences and emotions recorded in a journal kept by each student, and (3) an examination of one's personal characteristics in light of the various theoretical interpretations which have been made concerning possible reasons for test results. For example, a student may receive the following pattern of test results: moderate to high general self-esteem; moderate to low scores on social self-confidence measures, with high variability over time; high scores on achievement-oriented measures; and heavy weighting of intelligent-competence traits in the rating of acquaintances. This student may note that the low degree of relationship between general and social self-esteem reflects his or her feelings of general self-worth coupled with feelings of discomfort and unease in large groups of people. Thus, the question of the unitariness of self-esteem takes on personal meaning. The student may also note that work-related values are most important to him or her in judging self as well as others, and may reflect that the high general self-esteem derives primarily from feelings of intellectual competence.

In looking at the journal, the student may note a relationship between social self-esteem scores and experiences in relating to other people. This student may then go back and examine various theories about the role, origin, and effects of self-esteem in light of these patterns in his or her own data. Of course, there are cases in which test scores do not fit well with self-image. Here the student must reexamine both the test (Was a different interpretation placed on the question than that originally intended?) and the view of himself or herself.

It should be noted that while the teaching of methodology is not a purpose of this class, the course is an extremely effective way of teaching psychometrics and statistics. The students quickly learn to examine item wording and spontaneously demand item-test correlations for disputed questions, etc. Confronted with means, standard errors, and correlations and needing this information to interpret their own results, they learn to use it. Most students leave the class with at least a rudimentary understanding of factor analysis and understand the meaning of factor loadings and eigen values. As one student stated, "I feel like I got a pretty good pre-introductory course in statistics. The numbers mean something to me and that is an accomplishment of its own."

The Psychology of Yourself has been taught twice, to a total of forty-two students. With the procedures well-tested, the number of students participating in the class is limited almost solely by laboratory facilities. If the class exceeds fifteen, an opportunity for small-group discussions of test results should be offered. These groups can be led by graduate students or by students who took the course previously.

### Student Reactions

In general, the class is a joy to teach because the students get deeply involved in the material and love the class. They are willing to tackle complex theoretical ideas, and they continue to demand further explanation until they understand. Several of the students in the class contributed original theoretic ideas relevant to this writer's work on self-structure. Their suggestions for further areas of research were also excellent. Many of these students have continued to take independent-study credits, working on research with the individualized instruments. Several of them have asked to retake some of the measures after the course was over, in an attempt to assess changes in themselves.

We have stated that the students were enthusiastic about the course. Their own reactions may be the best evidence that CSI can indeed be used in teaching complex conceptual ideas and in helping students apply concepts to real-life situations.

> Thank you for one of the most meaningful and practically applicable classes I've ever had. I have so many more tools for thinking about myself now and for understanding and maybe even changing some of the ways I think.
>
> This class has gotten me to do a lot of thinking as to why I am the person I am... . The psychology of yourself has been of great value to me in that I've been directed towards working on insights and working through conflicts that would have most likely been pushed to the back of my mind otherwise.
>
> I discovered much about myself because I had never had a coherent framework in order to see myself with. I feel that this has been one of the biggest learning experiences of my college career.

The measures provided good intellectual stimulation and in many cases flushed out many thought processes that were either undeveloped or highly latent. In a sense they presented material that provided input into the question of what comprises this entity (me).

Even though we were probably quite alike in many respects, it was fascinating to observe all the differences in the people, and how sensitive they were to the differences. I guess that brings me to the main point I learned from this class—something I knew but easily forget. I must try always to be kind. People are so complex and so fragile.

### Additional Applications

In addition to continuing the present course, we will be using the PROCTR applications described in this paper in a personality theory course, an experimental personality methods course, and a course exploring awareness of group interactions. With expanded facilities, we would also use these applications in the teaching of large introductory social and personality psychology courses. In all of these courses, the use of PROCTR to administer psychological instruments, with the student receiving personal feedback, offers the potential of increasing self-awareness and at the same time providing information about personality testing.

PROCTR has been used in several research applications emphasizing features of the program in addition to those emphasized in the Psychology 400H application. A graduate student in psychology used PROCTR to record interaction ratings made by judges who viewed videotaped therapy sessions. The recording of the responses through PROCTR required less delay and less of the raters' attention than would have been required with other recording methods.

In another application, PROCTR was used in panel survey research to examine attitudes toward candidates and issues over the course of the 1976 presidential debates. Subjects for this study included party precinct delegates and covered a broad age range. These subjects were intrigued with the use of the computer as a data collection device and mentioned feeling at ease with the "computer as an interviewer" because of the anonymity of the responses.

In another study, PROCTR was used to present stimulus materials in a traditional, concept-learning task. The experiment was set up in a verbal discrimination learning format. Subjects were presented with two concepts, allowed six seconds to identify the correct item, and were then shown the correct answer for six seconds. Pairs were presented in a different random order for each subject and the responses were resorted into a single order for analysis. This is the first application of PROCTR which has made use of the timing features.

A broad range of additional instructional and research applications of the program are currently in the planning phase. Strong emphasis is being placed on the options which allow the administration of individualized instruments. One study of the psychological effects of life-threatening illness will involve use of a portable terminal and telephone lines to administer psychological instruments outside the laboratory setting. The most notable point about this set of applications of CSI is its flexibility. These procedures may be used for teaching, for research, or for some combination of the two; and the topics dealt with range from memory and verbal learning to a subject's own hopes and fears for the future.

### Notes

1. Time restriction options are available in PROCTR, but have not been used in the current application.

2. An exception is a test such as Bem's androgeny scale (Bem 1974) where the sub-scale sums are transformed into t scores using a measure of the subject's response variance.

### References

Bem, S. L., The Measurement of Psychological Androgyny. *Journal of Consulting and Clinical Psychology*, 1974, *42*, 155-162.

Kelly, G. A., *The Psychology of Personal Constructs*. New York: Norton, 1955, Vol. 1.

Price, L. A., *Automated Administration of Psychological Instruments: Pedagogical and Practical Considerations*. Computer Institute for Social Science Research, MSU, 1977. (A paper submitted for CCUC/8.)

# Computer-Managed Instruction Produces Better Learning in an Introductory Psychology Course

**James H. Roll and Robert M. Pasen**
*William Rainey Harper College*
*Palatine, Illinois 60067*

## Introduction

After a 1972 survey revealed that approximately 35 percent of the introductory psychology students at Harper College received failing or near failing grades on their final examinations, the authors began searching for a more effective teaching technique. A review of the literature uncovered a technique known as the Keller Plan or Personalized System of Instruction (PSI), which seemed to offer a possible solution to the problem.

The Keller Plan employs a form of "mastery learning" in which the student works at his own pace through a series of study units. He is required, however, to demonstrate "mastery" of a particular unit by obtaining a score of 80 to 90 percent on a unit quiz before proceeding to the next one (for a more detailed description of the actual procedures see Kulik, Kulik, and Carmichael, 1974, p. 379).

A seemingly essential element for the implementation of the Keller Plan is the student proctor. Proctors are responsible for assisting students in their studies, administering unit quizzes, grading quizzes, and discussing the results with the student. In addition, the proctor must keep progress files on each student. Because the responsibilities are numerous and time consuming, a proctor typically works with a maximum of eight to ten students. Without proctors it would seem to be extremely difficult to implement a course which incorporates mastery and self-pacing.

Unfortunately, at a community college such as Harper, rapid student turnover makes it difficult to find, train, and adequately supervise proctors. Further, restrictions on special course offerings and a lack of funds make it almost impossible to find a means of adequately compensating proctors.

Harper's acquisition of a computer for instructional purposes seemed to offer a solution to this problem. By having the computer perform many of the clerical functions of a student proctor (i.e. testing, grading, and record keeping), it seemed possible to retain the mastery and self-pacing features of the Keller plan without the use of proctors. Although the authors recognized that the elimination of human proctors might result in the loss of a certain amount of the personal attention and assistance which students typically receive, it was not clear whether such a loss would be detrimental.

The first step in the process of implementing a computer managed course was the acquisition of appropriate software. A review of several CMI programs such as the interactive version of TIPS developed by R. R. Roskos at the University of Wisconsin, LaCrosse, indicated that none of the then available software would completely satisfy the needs of the proposed course. Therefore, the authors decided to develop their own software. At the time the present study was conducted, the software consisted of the following nine programs:[1]

PSYTST: Allows the student to access the system, reads the student's record to determine which quiz he should take, and chains to the appropriate quzzing program.

PQ1-PQ10: (The actual quizzing program.)[2] The program makes up a quiz (by randomly selecting questions from the appropriate QUIZ file of forty-eight test items), administers the quiz to the student, scores it, provides the student with feedback (from the appropriate FDBK file) regarding concepts he does not understand, and gives him page references to portions of the text. The program also writes test item analysis data into the appropriate ANSW file. PQ chains to SCRIPT. (See sample output in Appendix.)

SCRIPT: Writes the student's score in his individual progress record contained in the RSLTS file. SCRIPT chains to DUMMY.

DUMMY: This program dumps the student out of the CMI system and thereby prevents him from tampering with any of the major system programs.

SKLKEY: A file management program which allows an unsophisticated user to load test and feedback items into appropriate files, as well as modify items, and make lists of the contents of the files.

LOAD: Allows the instructor to input class lists into a newly established RSLTS file.

DATUM: Allows the instructor to add, delete, or modify individual scores in a student's record, as well as erase an entire student record.

RSLTS: Provides the instructor with an updated printout from the RSLTS file. This includes a listing of each student in the class, displays the scores of all quizzes taken by the student, and the date upon which each unit was completed. If desired, a bonus point for each unit passed before its particular due date can be awarded automatically. This encourages the student to keep up with his work. As soon as the student has finished the course, RSLTS automatically totals the highest obtained score on each of the units. RSLTS also provides useful information such as (1) the mean and standard deviation of score totals, and (2) the mean and standard deviation of the number of trials required by the class to attain mastery on each unit, which is useful for evaluating difficulty levels. (See sample output in Appendix.)

ANLSIS: Provides the instructor with a detailed test item analysis for questions in the QUIZ file. For multiple choice and true-false items, it prints the number and percentage of students who selected each alternative response. For fill-in questions, it prints the number and percentage of students who answered the question correctly and makes a print-out of all answers which the computer accepted as "correct" and all responses which the computer rejected. This information helps the instructor determine how to adjust the "key lettering" of fill-in answers so that he can allow for acceptable misspellings.

In light of the first year of experience with the CMI software, one additional program (RESPON) has been added. Upon completing each unit quiz at a CRT, the student immediately executes RESPON from a nearby printer, thereby producing a hard copy of appropriate feedback information.

In addition, RESPON prints out a table with the following information: the record number of the item, the student's response to it, whether the response was correct or incorrect, and the student's reaction time in seconds. (See sample output in Appendix.) Prior to the development of RESPON students had to read their feedback messages from the CRT and write them down on paper. With RESPON both the students and the instructor save considerable time and effort. The information provided by RESPON is also valuable for resolving student disputes about the grading of quizzes as well as counseling students regarding their conceptual weaknesses.

## Purpose

The purpose of the study was to answer the empirical question of whether gains in final exam performance similar to those reported in the PSI literature could be attained by the use of a computer instead of human proctors. An additional purpose was to discover whether student attitudes towards a computer managed course would be at least as favorable as those towards a traditional lecture-discussion class.

## Subjects

Student signed up for one of two sections of introductory psychology without prior knowledge that different teaching techniques would be used in each section. These groups were designated "CMI" and "Traditional" to correspond to the teaching methods used. Originally the CMI group consisted of 27 students and the Traditional group consisted of 22 students. In order to assure that the precourse knowledge of the subject matter was the same in the two groups, the subjects were matched on the basis of a 100 question, multiple-choice comprehensive examination. As a result, the final sample consisted of 17 matched-pairs of subjects.

### Procedure

Students in the CMI group used as their text *Psychology: Explorations in Behavior and Experience* by Mednick, Higgins, and Kirschenbaum, and the accompanying behavioral objectives study guide by Pasen and Roll. In addition, each week CMI students attended 40 minutes of lecture-demonstration and spent 110 minutes studying, receiving individual help from the instructor, or taking quizzes at the computer terminals. The computer administered 10-question, randomly generated quizzes to each student. It also provided immediate feedback to the student regarding his score, listed all concepts that were not adequately understood, and referred the student to appropriate pages in the text for further study. Although students in this group were allowed to work at their own pace, each unit had to be "mastered" (i.e., no less than eight correct) before the student could proceed to the next unit.

Students in the Traditional group used the same text as the CMI group, but did not use the objectives study guide. In addition to attending 150 minutes of lecture each week, students in this group took a single 10-question manual quiz which consisted of items selected at random from the same question files used with the CMI group.

The Traditional group also received immediate feedback on its quizzes and could discuss problem areas with the instructor, but could not retake quizzes. Students in this group had to proceed to the next unit regardless of the score obtained on a given quiz.

At the end of the semester, students in both groups were given the same 100-question final exam, which earlier had been used as a pre-test for matching purposes. This exam consisted of ten multiple-choice questions for each of the ten units covered during the semester. None of the items used for quizzing purposes for either group appeared on the exam. The Kuder-Richardson estimate of the reliability for the exam was .91 (KR-20).

At the completion of the term, the students in the two groups also filled out a forty item attitude questionnnaire consisting of five-point rating scales assessing various dimensions of the class experience.

### Results

*Pretest:* The mean pretest score for the CMI group was 40.12 ($n=17$, S.D. $=5.28$). The mean pretest score for the Traditional lecture discussion group was 40.12 ($n=17$, S.D.$=5.16$). A $t$-test for matched groups indicated that there was no significant difference in the pre-course knowledge of psychology between the two groups ($t=0$, d.f.$=16$).

*Final Exam:* The mean final exam performance for the CMI group was 73.35 ($n17$, $S=15.72$), while that of the Traditional lecture-discussion group was 56.94 ($n=17$, $S=11.22$). A $t$-test for matched groups indicated that the difference of 16.41 points was highly significant ($t=63$, d.f.$=16$, $p<.001$).

*Questionnaire:* Because the questionnaires were filled out anonymously, it was not possible to isolate the responses for the seventeen matched pairs of subjects. Nevertheless, the resulting comparison of all twenty-seven subjects in the CMI group and twenty-two subjects in the Traditional group offers some interesting insights into student reactions to the use of computers in the classroom. For the sake of brevity, only six of the more relevant items are discussed in the present article.[3] The items as they appeared on the questionnaire are listed below:

1. In comparison with other courses I have taken, I feel that my *understanding* of basic concepts and principles in this course is
   a. much greater
   b. greater
   c. about the same
   d. less
   e. much less
2. Considering all of the qualities which are applicable, how would you rate this course?
   A B C D E
   Excellent Average Very Bad
3. In helping you to learn the course content, how would you rate the contribution of the *instructor*?
   A B C D E
   Very Helpful Neutral Very Unhelpful
4. In comparison with other courses I have taken, the amount of individual help I am able to get from my instructor in this class is
   a. much greater
   b. greater
   c. about the same
   d. less
   e. much less
5. In helping you to learn the course content how would you rate the computer quizzing system?
   A B C D E
   Very Helpful Neutral Very Unhelpful
6. The idea of taking my chapter quizzes at a computer terminal is
   A B C D E
   Very Pleasant Neutral Very Unpleasant

A summary of the students' responses to these items appears in Table 1.

As can be seen in Table 1, students in the CMI group felt they attained a better understanding of basic concepts and principles and gave the course a higher over-all rating. Although there was no difference between the groups in the extent to which they perceived the instructor as helping them master the course content, the students in the CMI group felt significantly more *able* to get individual help from the instructor. If the assumption is made that a rating of 3 on item #5 really constitutes a "neutral" rating, then the mean rating of 4.04 by the students in the CMI group means they perceived the computer quizzing system as "helpful." If a similar assumption is made about a rating of 3 on item #6, then the mean rating of 4.26 by the students in the CMI group means that they found the process of taking quizzes at a computer terminal "pleasant."

**Table 1. Attitude Ratings of the CMI and Traditionally Taught Groups**

| *Item* | *Group* | *No.* | *Mean* | *S.D.* | *t-Value* | *Significance Level** |
|---|---|---|---|---|---|---|
| 1. Understanding of basic concepts | CMI<br>TRAD | 27<br>22 | 3.50<br>3.50 | .67<br>1.14 | t=3.04<br>d.f.=47 | p<.01 |
| 2. Over-all course rating | CMI<br>TRAD | 27<br>22 | 4.52<br>3.68 | .85<br>.89 | t=3.35<br>d.f.=47 | p<.01 |
| 3. Contribution of instructor to learning | CMI<br>TRAD | 27<br>22 | 3.93<br>3.95 | .87<br>1.09 | t=.10<br>d.f.=47 | p>.05 |
| 4. Amount of individual help from instructor | CMI<br>TRAD | 27<br>22 | 3.81<br>2.77 | 1.00<br>1.06 | t=3.52<br>d.f.=47 | p<.001 |
| 5. Helpfulness of computer quiz system | CMI | 27 | 4.04 | 1.09 | | |
| 6. Pleasantness of taking quizzes from computer | CMI | 27 | 4.26 | 1.13 | | |

*all t-tests are 2-tailed

### Discussion

As previously mentioned, the CMI group obtained a significantly higher mean final exam score than the Traditional group. This finding is confounded, however, by the fact that different teachers taught the two groups. The fact that students in the CMI group did not perceive that they learned any more from their instructor than did the students in the Traditional group (see item #3 of Table 1) suggests that differential teacher effectiveness was not the critical factor in determining the difference between the two groups in final exam performance. It seems more likely that the superior performance of the CMI group was due to the greater effectiveness of the CMI strategy.

The responses to item #1 of the attitude questionnaire (see Table 1) indicate that members of the CMI group felt that they gained a greater understanding of basic concepts and principles than did the members of the Traditional group. Thus, the students in the CMI group not only learned more as indicated by the final exam results, but they also had a greater feeling of achievement.

An often heard criticism regarding the use of computers in education is that they depersonalize the learning process because the student is forced to interact almost exclusively with a machine and has little access to a human instructor. In the present study, however, the responses to item #4 of the questionnaire (see Table 1) suggest that students in the CMI group were able to get more individual attention from the instructor than students in the Traditional group. This is most likely due to the fact that the use of a computer to perform routine clerical duties such as quizzing, grading, and record keeping allows the instructor to spend more time in assisting students on an individual basis.

The responses of the CMI group to two of the questions suggest that the computer quizzing system was well received. More specifically, the responses to item #5 of the questionnaire suggest that this group found the computer quizzing system "helpful," while the responses to item #6 suggest that the CMI group found taking quizzes at a computer terminal to be "pleasant."

Finally, responses to item #2 of the questionnaire indicate that the CMI group gave the course a higher "over-all" rating than did the Traditional group.

In addition to suggesting that CMI leads to significant differences over traditional approaches in both the amount learned and the students' ratings of various aspects of the class experience, the data further indicate that CMI offers an alternative to the use of student proctors in individualizing the learning process.

*PSI: Are Proctors Really Necessary?*

The difference of 16.4 percent between the CMI and Traditional groups on their final exam performance compares quite favorably with differences reported between other traditionally taught introductory psychology courses and Keller's PSI technique. Such differences have been replicated using a wide variety of texts and course materials (for excellent summaries of the relevant PSI literature see Robin, 1974; Ryan, 1974, and Sherman, 1974).

The fact that this 16.4 percent difference was achieved without the use of proctors tentatively suggests that the effectiveness of PSI may not be due to the use of proctors, per se, but to some other aspect of the course (e.g., the use of objectives, mastery criteria, etc.). Although the present data do not permit precise determination of the effective element(s) of PSI, they do suggest that the functions which proctors perform can be carried out more reliably by a computer. This possibility should be of interest to instructors experiencing difficulties in obtaining, training, and/or compensating human proctors.

In addition to greater reliability in the quality of feedback provided to the student, CMI can also provide automatic test item analysis for each item which, when accumulated over a long period of time, increases the likelihood that an instructor will be able to make more valid decisions about which test items, answers, and/or objectives need modifying. Such a procedure is certainly less haphazard than the practice, found in typical PSI courses, of asking proctors to verbally report "troublesome" items to the instructor.

### Future Directions

Plans for the near future include the development of a test generating program that is compatible with our current software. Once this program is complete, any faculty member who creates an extensive set of test items could generate an almost infinite variety of test and quiz forms for classroom use as well as on-line quizzing. At the same time, a program will be developed to interface a mark sense scoring device with the present software, so that an instructor using paper and pencil tests can take advantage of the record keeping and test item analysis capabilities which exist in the system. The utilization of this test scoring program would enable an instructor to pool test item analysis data from both on-line quizzing and off-line testing, thereby increasing the data base for item analyses.

### Notes

1. The software described here was written in time-shared BASIC for a Hewlett Packard 2000E by J. Roll and J. Siedband of Harper College. The authors gratefully acknowledge the help of J. Siedband whose early input and continued consultations made possible the development of these programs.

2. Originally there were ten versions of this program, one for each unit quiz. With our current Hewlett Packard 2000 Access System there is only one program and, depending on which unit the student is being quizzed, different files are automatically assigned to it.

3. A copy of the original forty item questionnaire as well as the students' responses can be obtained from the authors.

### References

Ferguson, G. A. *Statistical Analysis in Psychology and Education*, 4th Ed. New York: McGraw-Hill, 1976.

Keller, F. S. A personal course in psychology. Paper read at the American Psychological Association Convention, Philadelphia, August 1963. Also in R. Ulrich, T. Stachnik and J. Mabry (eds.), *Control of Human Behavior* (Vol. I). Glenview, Ill.: Scott Foresman, 1966, pp. 91-93.

Keller, F. S. New reinforcement contingencies in the classroom. Paper read at the American Psychological Association Convention, Chicago, September 5, 1965. Also in J. G. Sherman (Ed.), *PSI: 41 Germinal Papers*, W. A. Benjamin, 1974, pp. 212-215.

Kulik, J. A., Kulik, C. L., and Carmichael, K. The Keller plan in science teaching. *Science*, 1974, 183, 379-383.

Mednick, S., Higgins, J., Kirschenbaum, J. *Psychology: Explorations in Behavior and Experience*. New York: Wiley, 1975.

McMichael, J. S. and Corey, J. R., Contingency management in an introductory psychology course produces better learning. *Journal of Applied Behavior Analysis*, 1969, 2, 79-83.

Pasen, R. and Roll, J., *Student Guide to Accompany Psychology: Explorations in Behavior and Experience*. New York: Wiley, 1975.

Robin, A. Behavioral instruction in the classroom: a review. Stony Brook, New York: State University of New York Instructional Resources Center. *Technical Publication No. C-1-74*, 1974.

Ryan, B.A. Keller's personalized system of instructon: an appraisal. Washington, D.C.: American Psychological Associaton, 1974.

Sherman, J. G. Application of reinforcement principles to a college course. Paper read at the American educational Research Association Convention, New York: February 17, 1967. Also in J. G. Sherman (Ed.), *PSI, 41 Germinal Papers*, W. A. Benjamin, 1974, pp. 216-219.

Sherman, J. G. PSI: an historical perspective. Paper read at the Rocky Mountain Psychological Association Convention, May 13, 1971. Also in J. G. Sherman (Ed.), *PSI: 41 Germinal Papers*, W. A. Benjamin, 1974, pp. 220-222.

Sherman, J. G. The theory behind PSI. Paper read at the West Coast PSI Conference, California State University, Long Beach, California: September 18, 1972. Also in J. G. Sherman (Ed.), *PSI: 41 Germinal Papers*, W. A. Benjamin, 1974, pp. 223-225.

Sherman, J. G. (Ed.), *PSI: 41 Germinal Papers*. W. A. Benjamin, 1974.

## Appendix

```
RUN
PSYTST

TEACHER'S NAME?ROLL
YOUR NAME?GORT

YOU SHOULD BE READY FOR A QUIZ ON CHAPT 1 & APP
DO YOU WANT TO TAKE THE QUIZ NOW (YES/NO)?Y

ANSWER ALL QUESTIONS BY TYPING YOUR ANSWER
IMMEDIATELY AFTER THE QUESTION MARK.

QUESTION # 1
TO GENERALIZE A RELATIONSHIP STATEMENT TO EVENTS OTHER THAN THOSE SPECIFICALLY
OBSERVED, IT IS MOST IMPORTANT TO A) SELECT REPRESENTATIVE SAMPLES B) VARY DEPENDENT
VARIABLES C) USE A DOUBLE BLIND CONTROL

ANSWER?A

QUESTION #2
A MAJOR ADVANTAGE OF NATURALISTIC OBSERVATION OVER THE EXPERIMENTAL METHOD IS
THAT IN THE FORMER, ENVIRONMENTAL EVENTS CAN BE MORE EASILY MANIPULATED.
A) TRUE    B) FALSE

ANSWER?A

QUESTION #3
AN OLD TELEVISION COMMERCIAL INSISTED THAT 'ATHLETES ARE MADE, NOT BORN.' THE
MANUFACTURER WAS OBVIOUSLY TRYING TO CONVINCE PEOPLE THAT ATHLETIC ABILITY WAS A
FUNCTION OF AN ORGANISM'S A) NATURE   B) NURTURE

ANSWER?B
```

```
QUESTION #10
WHICH OF THE FOLLOWING IS NOT AN EXAMPLE OF BEHAVIOR? A) WALKING B) FIXING A CAR C)
THINKING ABOUT DINNER D) WRITING A POEM

ANSWER?C


DO YOU WISH TO DOUBLE CHECK ANY QUESTIONS (YES/NO)?Y
WHICH ITEM NUMBER?2


QUESTION #2
A MAJOR ADVANTAGE OF NATURALISTIC OBSERVATION OVER THE EXPERIMENTAL METHOD IS
THAT IN THE FORMER, ENVIRONMENTAL EVENTS CAN BE MORE EASILY MANIPULATED. A) TRUE B)
FALSE

YOUR FIRST ANSWER WAS: A
WHICH ANSWER DO YOU CHOOSE NOW   B


DO YOU WISH TO DOUBLE CHECK ANY QUESTIONS (YES/NO)?N

  9          CORRECT OUT OF        10

GREAT! YOU ONLY MISSED ONE. GO TO THE NEXT UNIT.
```

```
GET-RESPON
RUN
RESPON

THIS PROGRAM DOES ANY OF THE FOLLOWING:
      1)MAKES A PRINT-OUT OF THE QUIZ RESPONSES OF THE STUDENT SPECIFIED BY THE
        PROGRAM USER.
      2)MAKES A PRINT-OUT OF THE QUIZ RESPONSES OF EVERY STUDENT IN THE FILE.
      3)ERASES THE QUIZ RESPONSES OF ALL THE STUDENTS IN THE FILE.
WHICH OPTION# DO YOU CHOOSE? 1

STUDENT'S NAME?GORT

GORT                                                                  JAN 10    1977

                                       QUIZ 1A

    ITEM #          REC.#      YOUR ANS.               SCORE      RESP. TIME (SECONDS)

      1              15        A                         1                12
      2              30        A                         0                 6
      3               1        B                         1                 6
      4              42        SCATTER                   1                13
      5              44        HIGH                      1                14
      6              18        B                         1                32
      7              36        DESCRIPTIVE               1                20
      8              45        B                         1                40
      9              20        DEPENDENT                 0                12
     10               4        C                         1                29

ITEMS REVIEWED

      2              30        B                         1                 4

 9      OUT OF 10      CORRECT
ELAPSED TIME =5      MINUTES

YOU MISSED ITEM #9          REC # 20

YOU WERE NOT ABLE TO RECOGNIZE AN EXAMPLE OF AN INDEPENDENT VARIABLE  SEE PAGE 533
OF THE TEXT.

STUDENT'S NAME?
STOP
```

```
RSLTS
                                DEC 22. 1976

4) DORIS BERO
CH 10          10                                          10 OCT  13
CH 11           9                                          10 OCT  21
CH 2            9                                          10 OCT  28
CH 3            7            10                            10 NOV   9
CH 4           10                                          11 NOV   9
CH 8            9                                           9 NOV  23
CH 9            8             9                            10 NOV  23
CH 13&14        7             9                            10 DEC   2
                                   TOTAL=70
                             70 OUT OF 70 POINTS= A
```

```
5) LYNN BICKNASE
CH 10          10                          10 OCT 13
CH 11           4        8                  8 OCT 21
CH 2            7        8                  8 NOV  2
CH 3            5        8                  8 NOV 15
CH 3            7                           7 NOV 18
CH 8            8                           8 DEC  2
CH 9            6        10                10 DEC  7
CH 13&14       10                          10 DEC  9
                               TOTAL=59
                     59 OUT OF 70 POINTS= B
```

```
              TOTAL           MEAN
QUIZ #        # OF TRIALS     # OF TRIALS     S.D.

Ch 1&APP          4              2.00         1.00
CH 6              4              2.00         1.00
CH 10            34              1.00         0.00
CH 11            59              1.90         0.73
CH 2             48              1.60         0.66
CH 3             47              1.62         0.72
CH 4             43              1.48         0.62
CH 8             52              1.79         0.85
CH 9             56              1.93         0.78
CH 13&14         46              1.64         0.77
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
********
TOTALS          393              1.62         0.75

MEAN TOTAL POINTS= 64.64
STANDARD DEVATION= 5.41
```

# Automated Administration of Psychological Instruments: Pedagogical and Practical Considerations

**Leighton A. Price**
*Computer Institute for Social Science Research*
*Michigan State University*
*East Lansing, Michigan 48824*
*(517) 353-2040*

## Introduction

In the College of Social Science at Michigan State University, we have found that automated administration of psychological instruments has considerable pedagogical value. The principle course in which instrument presentation software has been used is described in a paper by Thompson.[1] This course is called "The Psychology of Yourself" and owes its existence to the availability of this software. The course was offered last winter term and again this fall.

This paper examines: (1) Our rationale for venturing into this mode of instructional support, (2) the structural characteristics and specific features of hardware and software used by instructors, secretaries and students, and (3) current or projected applications which further demonstrate the value of automated instrument presentation. The paper deals with some matters which are broader than those discussed by Thompson. It also presents more detailed information concerning the hardware and software employed.

## Problems Facing Computer-Supported Instruction (CSI) at Universities

The university environment encourages diversity of thought and innovative approaches to problems. There are also expectations that creativity be exercised within some relatively narrow area of specialization. As a result, instructors have both the opportunity and the obligation to teach courses in ways that impart some of their own special knowledge and skills.

This situation poses some perplexing problems for those involved in developing instructional uses of computers. Essentially, it implies that extensive and rigorously developed CSI materials are unlikely to receive wholesale adoption by an instructor. It implies that there must be easy ways to tailor materials to meet the special needs of individual instructors.

An additional problem is that the distinction between instruction and research is sometimes difficult to identify in undergraduate courses. At the graduate level this situation is especially prevalent. For example, there are courses on research methods, test construction, analysis of cases or situations, observation techniques, etc. As a result, it is important to have CSI materials which function well in both domains.

## Partial Solutions to the General Problems

In the summer of 1974, the Computer Institute for Social Science Research established a Behavioral Science Instructional Laboratory (BSIL) at Michigan State University. The main purpose of the laboratory is to help instructors utilize compterized techniques to augment undergraduate courses.

BSIL was established when we received an NSF scientific equipment grant[2] for interactive computer terminals and a grant from the Ford Venture Fund[3] to provide a first-year operating budget for the laboratory.

During the first year we used six, and later eight, Texas Instruments (TI) 733 printing terminals. These terminals were used via a dial-up timeshared system on the University's

CDC6500 computer. Initially, we concentrated on augmenting social science research methods courses, and approximately 1,000 students used BSIL.

In our second year of operation, we switched to a timeshared Hewlett-Packard (HP) 2000 system capable of supporting thirty-two terminals simultaneously. At the time the HP2000 was ordered, we made arrangements to merge terminal use planned by journalism with our uses of BSIL. The laboratory now has our eight TI733 terminals and two HP2640 video display terminals plus four HP2640 terminals purchased by the School of Journalism with funds from a Gannett Foundation grant.[4] All of these terminals are used with the HP2000 computer over hardwired cables. BSIL also obtained an Advent video projector which works off a terminal with composite-video output. The Advent is used in training demonstrations for groups of students.

During the past year, the BSIL terminals were used by approximately 2,000 social science students. Most of the students were in research methods courses, but a number of substantive courses were also represented. This year we anticipate serving about the same number of students. At the same time, we are gradually extending the variety of courses we serve and are exploring instructional uses that involve more intensive interaction with students.

Recently we placed an order for several more HP2640 terminals. These terminals will improve our opportunities to serve instructors who have expressed interest in using BSIL but felt that we had too few video display terminals to meet their needs.

## Automated Presentation of Psychological Instruments

Automated presentation of psychological instruments is one of the new modes of instructional support with which BSIL is involved. For this mode we are asking instructors to prepare CSI materials for their courses and to become involved in the process of entering the material. BSIL's major role here has been to develop computer programs which make it easy for faculty and students to employ this mode of instructional computing. The programs are also designed so that instructors can utilize existing secretarial personnel when preparing materials.

In the past we have prepared all the materials used to augment courses (i.e., software, data bases, demonstrations, CAI lessons, etc.). The instructors' responsibilities were to meet with us and decide exactly how computer use could be coordinated with a course and to reorganize course content where necessary to accommodate these experiences. In addition, we conducted the training demonstrations for students and had laboratory assistants available whenever the laboratory was open. For automated presentation of instruments we are attempting to minimize our direct involvement.

From a computing point of view, automated presentation of instruments constitutes a rather unsophisticated application. From an instructional support viewpoint, however, this represents a significant new direction for BSIL.

## Specific Circumstances Dictating Our Moves

Some of the circumstances which led us to think that automated presentation of instruments had considerable promise are discussed below.

1. Students in the social sciences seldom have much experience with computing equipment of any kind. They feel intimidated by technology and shy away from learning computer programming. However, such individuals readily find the computer a useful tool when they work interactively with applications that minimize their need to have technical knowledge about computers.

2. Faculty in the social sciences seldom have strong background in computer use and are not prepared to acquire extensive computer expertise as a prerequisite for developing instructional computing materials. Consequently, we need applications and support software which recognize the seriousness of this problem. Until we do, the instructors

who are determined to utilize modern technology to augment classroom experiences will inevitably find that they become committed to more than they had anticipated.

3. In the university environment, instructors are generally provided with the secretarial and clerical support needed to prepare instructional and test materials for students. When instructors want to develop instructional materials for use on computers, however,the existing staff is generally of little or no use, and requests for hiring computer programmers tend to fall on deaf ears. If the level of computer expertise required for developing instructional computing materials can be reduced, available secretarial and clerical personnel could help and use of CSI is more likely to grow.

For instructional uses of computers to be successful in the social sciences, it must be easy to make use of instructional software or prepare materials for use on a computer. We should not aim for drastic alterations in established behavior or expect people to acquire complicated new skills. In the social sciences, the computer is still largely unfamiliar territory. If we cope with these circumstances realistically, the potential for instructions applications of computers is enormous. Failure to recognize the existence of these circumstances could eventually spell disaster for CSI in the social sciences.

### Preparation and Presentation of Instruments

An overview of the software developed by BSIL and the hardware used for this application follows. The software is written in BASIC for our HP2000 computer.

To enter psychological instruments into the computer, we use a word processing program called SCRIBE. This interactive routine is extraordinarily easy to use since its main purpose is to transfer blocks of text to and from the memory of a video display terminal and to maintain updated text files. The bulk of the program is concerned with informing people what terminal key to press next and with protecting them against possible mistakes. A person using the program simply types text as it would be typed on a sheet of paper. There are no word processing commands to be learned. The basic intent of SCRIBE is to take full advantage of the storage and editing capabilities of sophisticated video display terminals such as our HP2640's.

The few editing keys on an HP2640 are easy to use, and changes, insertions or deletions are immediately seen on the screen. As a result, the process of entering psychological instruments is convenient, and it takes only a few minutes to train a typist to use the program.

The program which presents psychological instruments at a terminal is called PROCTR, and it is the main piece of software developed for this instructional application. This program is designed to retrieve the text files for instruments and administer them. Since instruments are structured in a generic form for entry and are then compressed to reduce disk storage, the program unpacks instruments and administers them according to specifications contained in the instrument files themselves.

As an instrument is being presented, PROCTR checks responses for type and range. Students are also required to answer all questions. Consequently, the data are especially clean.

After an instrument has been completed, a student's responses are transferred to a file containing data for completed instruments. If a student stopped prior to completion, the program will insure that the instrument is completed before it adminsters another one to that individual. The combined response file is stored in a format which facilitates transferring it to the CDC6500 for possible analysis.

Student use of PROCTR is facilitated by a supervisory program called the Instructional Delivery System (IDS). This system brings students directly into the software which an instructor has decided to use in a course. The student simply logs into the computer using the authorized identification for a class and control is automatically transferred to the appropriate software. IDS is also capable of printing bulletins to all students in a course.

The instructor's role with IDS consists of specifying which software is to be available for student use, the account on which each piece of software is located, and the range of dates that each should be available for use by that class. These specifications are entered interactively and the process is prompted so it is very easy for an instructor to do. The instructor may also prepare new messages or bulletins for the class.

### Video Display Terminals

When instruments are prepared using SCRIBE, it is necessary to use video display terminals such as our HP2640 terminals. These terminals have versatile editing keys and memory for storing text as it is entered or edited. These features make it possible to conveniently do some rather complex editing without having any text editing commands. They also allow the terminal to be used in a manner which minimizes the demand on computer resources. The terminal is essentially running off-line until text is retrieved or entered in a block.

PROCTR may be used from either the TI or HP terminals, but it is most convenient to use the latter. There is no need to have paper output since responses are automatically stored.

The character set on these terminals is extremely legible, and students have found it easy to work at these terminals for relatively long periods of time.

### Organizing Instruments for Entry

When we began working on the problem of automating the administration of psychological instruments, we noticed a number of recurring features and problems. Some of these pertained to the repetitive characteristics of instruments, others to deficiencies in how printed instruments were usually administered. We concluded that most instruments could be conveniently defined in terms of more generic characteristics. Also, the administration of instruments could be more efficient and more rigorous when instruments were defined this way.

Following is a description of the component used to define instruments. Sometimes, all are required; at other times, only a few are needed.

1. Specifications Section. Once the main characteristics of an instrument have been identified and procedures for administering it have been decided, a specifications section may be prepared. The information in this section exercises much of the control over administration of an instrument.

2. General Instructions. Typically, an instrument begins with a set of instructions for the respondent. This may be straight text or include examples.

3. Response Instructions. When items have common response categories and/or common response scales, this information needs to be prepared only once. If instruments with common response instructions also require several types of responses per item, a response instruction section may be entered once for each type of response.

4. Headings Section. When the same set of items is repeated under different conditions, the headings or instructions defining each condition can all be entered together.

5. Items Section. The unique portions of items are entered in this section. For each heading the set of items is readministered.

Within each secton a number of directives are available to control the presentation of instruments. The primary commands for the specifications section are described below.

SPECIFICATIONS

INSTRUMENT NAME = ____________

(Any desired one-line identification)

INSTRUMENT ID = ______________

(A six-character identification which should also be used as the file name for the instrument)

RESPONSE TYPE = STRING
or
INTEGER. ______ . ___

(The string specification is for one-line open-ended responses. The integer specification is for numeric responses falling within the range indicated after the keyword. Negative numbers may be used.)

RESPONSES PER ITEM = ____________

(This indicates the number of answers required for each item. Multiple responses may be used regardless of whether items have common response instructions.)

RANDOM ORDER

(Items are presented in a random sequence but responses are stored in the order that items appear in the instrument.)

SUM = ____, ____, ---, ____
or
ALL

(The responses to items listed here are added together. More than one such specification may be given, but the option applies only for integer types of responses. These sums are transferred to the response file for all respondents regardless of whether individual responses are saved.)

CHAIN FILE = ______________________

(A six-character file name is given if this instrument must be presented in conjunction with another one.)

Note: Other commands pertain to limiting response times on items or instruments, saving response times, presenting headings in random order, and others.

The commands described below may be used in the general instructions section.

GENERAL INSTRUCTIONS

(Text which follows this command is presented prior to the first question.)

WAIT

(If this command is used, presentation halts after each screenful of text or as soon as all this text is presented. The instrument is restarted by pressing the return key.)

CLEAR

(The screen of the terminal is cleared before further text is displayed.)

LOCK

(Text already displayed is protected so that it will remain on the screen as other text is presented beneath it.)

UNLOCK

(A protected area becomes unprotected.)

Note: The above four commands may also be used with response instructions, headings, and items. The latter two directives operate only on our HP2640 terminals.

The primary directives for the response instructions section are as follows:

RESPONSE INSTRUCTIONS

(Response instructions common to all items are entered following this command.)

RESPONSE INSTRUCTIONS = ________

(When all items have a set of common response instructions, but there are several responses per item, separate instructions may be entered for selected ones or for all of them.)

When items are negatively worded, it is often convenient to include a command which causes the response to be recorded before it is saved.

REFLECT

(Whenever this command follows an ITEM command it indicates that the response value should be reflected before response values are stored. It applies only for integer responses.)

### Entering and Correcting Instruments

Over the past few years, we have used several different word processing machines. One of these machines had truly remarkable editing capabilities. Unfortunately, two weeks of rather intensive training guaranteed little more than marginal proficiency simply because it had so many capabilities.

We have also had experience with a much simpler word processing machine. This one is used very much as an ordinary typewriter. Nevertheless, it requires some careful training to use its cassette tape. Anyone who has difficulty remembering how to use machines is quickly overwhelmed.

When our secretaries used SCRIBE they found it so much easier to use that they didn't miss the sophisticated editing commands. In fact, the capabilities of the video terminals generally offset the need for most editing commands.

To date, dozens of types of psychological instruments have been put on our computer and a number of different people have been involved in entering and editing text. In all cases, it has taken only a few minutes to train people to use SCRIBE, and they are able to prepare instruments more easily than at a typewriter. More importantly, faculty wanting to help prepare instruments have found it very easy to use SCRIBE. They have not been shut away from the preparation process by equipment which is the special province of the secretary.

### Student Use of Instruments Administered by a Computer Terminal

Although students were initially apprehensive about using computer terminals, we found that they quickly adjusted to this new medium once it had been demonstrated to them. In fact, we found that students could be taught what they needed to know about using the terminals and about using PROCTR in approximately fifteen minutes.

The training demonstration for this application can be brief because PROCTR is programmed so it is virtually impossible for students to make mistakes that require them to seek help. Once a student is in PROCTR, both the operation of PROCTR and the administration of instruments are controlled by answers to questions. As a result, students are able to use PROCTR with confidence even though they know virtually nothing about computers.

It also has been our experience that students are interested in answering psychological instruments at computer terminals. The method is convenient for them and they have few reservations about giving frank, introspective answers. Arguments that instructional use of computers is impersonal and offensive are inappropriate in this instance. In fact, this approach even has some characteristics of a dialogue with the student.

For this type of instructional application, the computer terminal clearly has many advantages over paper and pencil administration.

### Cost-Effectiveness of Automated Instrument Presentation

On the surface, it may appear that the cost-effectiveness of CSI applications can be readily evaluated. Unfortuantely, there are many intangibles involved such as the quality of instruction,the value of reducing the time required to learn something, the importance of augmenting classroom experiences to meet different learning styles, and many others.

Some might argue that a lot of computer equipment is being used for a remarkably simple task. To judge the merit of this argument, however, it is also necessary to examine alternative approaches. Since we are dealing with new ways to augment instruction, there is little basis for comparison. The approach must stand largely on its apparent value to instructors.

From the comments below it should be evident that this approach to CSI is versatile and easily used to augment many types of classroom situations.

1. A broad, and easily altered, array of psychological instruments may be made available for instructional use.

2. The time and effort needed to prepare instruments and the disk space needed for storage are minimized by this approach.

3. Some major instructional uses of instruments can be entertained because the time dimension is substantially compressed in all phases of this approach.

4. Instructors can give students convenient access to instruments without incurring any supplies expenses, and departments are not buried in last minute mimeographing requests.

5. Students can receive more intensive training without also being burdened by time-consuming details that usually accompany doing more things in a course.

Where PROCTR has been used for research projects, the effectiveness of the approach is more obvious. In this case, the approach may be compared with alternative means of accomplishing the same thing. Compared with paper and pencil administration of instruments and with traditional data preparation techniques, the following advantages have been apparent in research applications of PROCTR.

1. It is easier and quicker to prepare instruments for use.

2. The costs for materials and labor to make copies of instruments are eliminated.

3. The difficulties of making sure that the proper people have the correct instruments are substantially reduced.

4. Since instruments must be completed and PROCTR insures that responses are appropriate, the data are exceptionally "clean." This can substantially reduce costs in the long run.

5. If items are administered in random order or response scales are reflected, PROCTR automatically stores all data in the appropriate order. It is unnecessary to code data so these conditions are recognized and write complicated software to reorganize the data.

6. The usual costs of coding, keypunching, and verifying data are eliminated.

7. Data are automatically sorted in a format which allows them to be sorted or otherwise processed with the statistical software available on large computers.

## Conclusions

The utility of the software described in this paper is very broad. During the past year it has been used for a number of purposes beyond the course mentioned here. In each instance, the application would have been more difficult or even unfeasible with traditional methods. This breadth of use is especially important in the university environment where instruction and research often overlap. The approach described here is clearly suited to problems in both domains.

From this paper it should also be evident that other disciplines besides psychology can make effective use of this mode of CSI. The structure we have used for defining psychological instruments is so general and so versatile that virtually any type of instrument can be defined this way.

The advantages of a structured approach to preparing psychological instruments are compelling. In fact, we recommend preparing instruments this way even when they are to be printed. A computer terminal with a high quality printing mechanism can be used to prepare masters from which mimeograph stencils or multilith plates can be made.

Before closing this writer wants to comment briefly on the importance of modern computer technology to the type of CSI described in this paper. If it were not for the availability of relatively inexpensive timeshared minicomputers, the application would have been difficult to justify and more difficult to implement. Computing can no longer be regarded as the province of data processing departments or of scientists writing applications for large computers. Our technology has evolved to the point that the kind of application described here is feasible and should be encouraged.

## References

1. Thompson, E., *The Role of Computer Supported Instruction in a Course on the Psychology of Oneself.* Computer Institute for Social Science Research, MSU, 1977 (A paper submitted for CCUC/8).

2. Price, L., Katz, M. and Prochera, J., *Instructional Laboratory Facilities for Undergraduate Research Methods Courses.* Computer Institute for Social Science Research, MSU, 1974. (A project supported by the NSF Instructional Scientific Equipment Program, GZ-3336.)

3. Price L., Katz, M. and Prochera, J. *The Behavioral Science Instructional Laboratory: A Facility for Upgrading Undergraduate Education.* Computer Institute for Social Science Research, MSU, 1974. (A project supported by the Ford Venture Fund.)

4. Hough, G., *A Proposal in Support of Journalism Education at the School of Journalism.* Michigan State University, Spring 1974 (Grant funded November 1974).

# An Elective Laboratory Course in Minicomputers in Chemistry

**G. E. O'Brien**
**M. A. Goldstein**
**A. C. Hayman**
**P. E. Sturrock**
*School of Chemistry*
*Georgia Institute of Technology*
*Atlanta, Georgia 30332*
*(404) 894-4012*

## Introduction

An elective laboratory course in analytical chemistry was introduced several years ago in the School of Chemistry, Georgia Institute of Technology. This course was designed to give advanced undergraduates and beginning graduate students an opportunity to study applications of minicomputers for data acquisition and instrument control in chemical laboratories. A second purpose of the elective course was to obtain experience in teaching these concepts and to develop experiments which might be suitable for inclusion in required courses such as the two-quarter sequence in analytical chemistry for chemistry majors.

The course consists of a one-hour lecture and six hours of laboratory each week for a quarter (ten weeks). The laboratory work is divided into four segments. Since most of our students have no previous practical experience in electronics, the first segment of the course consists of several simple operational-amplifier circuits. The second segment of the course is an introduction to programming in BASIC. Two minicomputers are available for this, a Digital Equipment Corporation (DEC) PDP-5 using a modification of DEC's 4K BASIC[1] and a Lab PDP-8e which is also used for all of the subsequent experiments with laboratory instruments. Students may do either segment one or two first, but must do both before moving onto the third segment.

The third segment of the course consists of several experiments in which the Lab PDP-8e is used for data acquisition from instruments operated in the normal way. For example, the computer is connected in place of or in parallel to the strip-chart recorder usually used for read-out on a Basic Carle gas chromtograph. DEC's data acquisition program, DAQUAN,[2] is used to store and display the chromatograms and to measure heights and areas of peaks, retention times, etc. A series of chromatograms with increasing sample sizes can be studied to show that peak area is superior to peak height as a measure of amount of component. Similarly the advantages of the computer are clearly evident to the student when working up infrared spectra obtained by the computer from the retransmitting slidewire of a Perkin Elmer model 337.

The fourth segment of the course consists of one or more experiments in which the computer controls the experiment in some way. For example, a motor driven buret can be controlled by the relays on the computer and potential data obtained by the computer from a pH meter. Programs can be written for automatic titrations using a variety of techniques to detect the end point. Another experiment in this fourth segment makes use of the digital to analog converters of the display control to control an operational-amplifier potentiostat. Various forms of polarography can be performed in this manner. Still another experiment is the automation of a UV-visible spectrophotometer. This experiment is discussed below in some detail.

### Additions and Modifications of Computer Hardware

The standard configuration of the Lab PDP-8e includes an 8 K memory, Teletype, high-speed paper tape reader and punch, display oscilloscope, 10-bit analog to digital converter (ADC) with 8-channel multiplexer, and a programmable real-time clock. In addition our system has an XY recorder connected in parallel to the display scope, 4 reed relays, a preamplifier with programmable gain, and a stepping motor with interface.

Each channel of the multiplexer has a preamplifier which originally had an input range of ± 1 volt. In some cases it was necessary to attenuate a signal to this range and in so doing the signal to noise ratio was degraded. Therefore the reamplifiers of four of the multiplexer channels were modified so that two of them now have an input range of 5 volts and the other two ± 10 volts.

### Computer Software Modifications

Due to the shortness of the course and the usual enrollment of fourteen to eighteen students, there is insufficient time for students to do any extensive programming in assembly language. Since we do not have the hardware items necessary to support an operating system, the use of a compiler based language was also ruled out. Thus, to achieve the objectives of the course, it was decided to use an interpreter type language and DEC's BASIC/RT[3] was selected.

As originally supplied, BASIC/RT had provisions for a user coded command, UCOM. The UCOM allows an assembly language routine, written by the user, to be inserted into core memory and called from BASIC/RT. This procedure is analogous to that suggested by Perone.[4] However, it was soon evident that we often needed to use more than one of our special devices in one program and this could not be accommodated with one UCOM. It therefore became necessary to modify BASIC/RT to include commands for the operation of the stepping motor, relays, programmable gain preamplifier, and recorder pen lift. We named this modified program BASIC/RT-GT.

Due to the shortness of the course and the usual enrollment of fourteen to eighteen students, there is insufficient time for students to do any extensive programming in assembly language. Since we do not have the hardware items necessary to support an operating system, the use of a compiler based language was also ruled out. Thus, to achieve the objectives of the course, it was decided to use an interpreter type language and DEC's BASIC/RT[3] was selected.

As originally supplied, BASIC/RT had provisions for a user coded command, UCOM. The UCOM allows an assembly language routine, written by the user, to be inserted into core memory and called from BASIC/RT. This procedure is analogous to that suggested by Perone.[4] However, it was soon evident that we often needed to use more than one of our special devices in one program and this could not be accommodated with one UCOM. It therefore became necessary to modify BASIC/RT to include commands for the operation of the stepping motor, relays, programmable gain preamplifier, and recorder pen lift. We named this modified program BASIC/RT-GT.

### Operation of a Cary 16 Spectrophotometer Under Computer Control

One of the first control experiments attempted was the operation of the Cary 16 UV-visible spectrophotometer. It has proved to be a particularly instructive project and each successive year the hardware and software have been improved. It is discussed here to illustrate the type of problems that students work with in the course. We do not consider the present arrangement to be the optimum way to automate this instrument, but rather as an effective way to teach concepts of computer automation with the limited equipment at hand, and at the same time to greatly increase the usefulness of the spectrophotometer.

The Cary 16 is a spectrophotometer with a double-prism monochromator and quartz optics covering the range from 160 to 800 nm. A single photomultiplier tube is used with both a tungsten-halide lamp and a deuterium lamp. A mirrored 60 Hz chopper is used to split the beam with the second beam being used to control the voltage to the dynode of the photomultiplier tube.

The usual mode of operation of the instrument is similar to that of the Beckman DU. First the wavelength is set, then the zero absorbance is set with the blank in the beam. Next the sample is positioned in the beam and the readout meter is again brought to null by adjusting a control. Finally the absorbance or transmittance is red from the control dial. The instrument is capable of very high accuracy and precision.

The first step in the automation project was to control the wavelength. Fortuitously, the stepping motor has 200 steps in each shaft rotation and the drive shaft to the wavelength control of the instrument changes the wavelength 20 nm in each rotation. Thus each step of the stepping motor corresponds to 1 angstrom unit. A mounting flange and coupling shaft were fabricated to mount the motor on the end of the monochromator housing and to couple it to the wavelength drive mechanism.

It was found necessary to limit the rate of rotation of the motor or else it would jump synchronization on starts, stops, and reversals. The maximum rate that can be used in each direction and at any wavelength is one step every 6 milliseconds. A stepping motor with a higher torque rating could undoubtedly be operated at a higher rate. However, the present slow rate of 16.7 nm per second is adequate for most purposes.

Examination of the electronic circuits of the Cary 16 revealed a convenient "test point" at which the amplified output signal from the photomultiplier was readily available. The voltage of the chopped signal at this point ranges from approximately −2.5 volts for the dark current to approximately 9.5 volts for 100%T. At first this signal was divided with a resistance network and fed into the ± 1 volt input of a multiplexer channel. After modification of the multiplexer preamplifiers, the unattenuated signal was connected directly to one of the ± 10 volt inputs.

In addition to the wavelength and the data signal line discussed above, a timing signal is required so that the computer samples the data line at the appropriate time in the chopper cycle. This signal was derived from one of the secondaries of the power transformer of the spectrophotometer.

It was found that the dark current varies slightly with wavelength and also with variations in sample absorbance. Therefore, it was necessary to obtain a value of the dark current for every value of the light current (either blank or sample). Fortunately, it is possible to obtain both of these values in a single 60-Hz cycle of the chopper. The light current is sampled immediately following the timing signal. The dark current is sampled after a delay timed by the computer clock.

Several programs have been written in BASIC/RT-GT to operate the instrument via the three interconnecting lines discussed above. Percent Transmittance, and absorbance can be obtained while stepping the wavelength in either direction. These spectra can be differentiated, but time is saved if derivative spectra are taken directly since it can be assumed that the base line is constant enough that it cancels in the derivative spectra and therefore need not be recorded. It was also easy to program the instrument to obtain repeated values at one or more wavelengths for kinetic studies. In addition, various computations can be performed on the data once they are in the computer.

Figure 1 shows the Teletype dialogue for the recording of an absorbance spectrum of 0.25 M $Ni(NO_3)_2$ from 350 to 750 *nm* with a data point every 2 *nm*. Additional spectra can be run using the same base line or a new base line can be run for each spectrum. The display oscilloscope is used to monitor the base line and then to display the spectrum as it is obtained, point by point.

In Figure 1, the 100 cycles means that at each wavelength, 100 values each of dark current and light current are obtained for averaging. Thus each absorbance value is the result of 400 individual measurements. Even with 100 cycles, the process takes only about 2 seconds per point.

The upper portion of Figure 2 shows absorbance spectra of $Ni(NO_3)_2$ (curve 1 = 0.25 M, curve 2 = 0.125 M). The lower portion of Figure 2 contains a point by point plot of the ratios of curve 1 to curve 2, versus wavelength. Considered as a set, these points average 2.032 ± 0.043 as compared to the theoretical value of exactly 2. However, the plot clearly shows the non-random nature of the error, which is greatest in the wavelength region of lowest absorbance. In the region from 490 to 520 nm, the absorbance of curve 2 is less than 0.01. Using the theoretical value of 2 for the ratio and assuming the values of curve 1 are correct, the worst error in curve 2 corresponds to 0.00065 absorbance units.

Figure 3 contains plots of values calculated from the data of Figure 2. The logarithmic form of the Beer-Lambert Law is:

$$\mathrm{Log}\,A = \mathrm{Log}\,a + \mathrm{Log}\,b + \mathrm{Log}\,c \qquad (1)$$

Since the only difference between curves 1 and 2 of Figure 2 should be due to concentration, curves 1 and 2 of Figure 3 should be the same except shifted by 0.301. A point by point subtraction of the two curves gives a value of 0.308 ± 0.011. However, as in the case of the absorbance ratios, the error is not random.

As suggested by Flaschka,[5] equation (1) readily leads to:

$$\Delta\,\mathrm{Log}\,A/\Delta\lambda = \Delta\,\mathrm{Log}\,a/\Delta\lambda \qquad (2)$$

The identical Δ Log A versus wavelength curves should result regardless of concentration or cell thickness. The lower portion of Figure 3 is such a plot derived from both curves 1 and 2 of the upper portion of the figure, plotted together. Once again the scatter is greatest in the region of low absorbance

### *Quantization Effects*

While making large scale plots of spectra obtained with the instrument, a stair-step effect was observed. this quantization effect results from the limited number of levels in the 10-bit ADC. While we were studying this problem, a paper by Horlich[6] appeared which advocated adding random noise to the signal and then averaging repeated measurements. The results of this procedure are equivalent to using an ADC with more bits.

Figure 4a shows the % T spectrum of 0.5 M $Ni(NO_3)_2$ from 530 to550 nm. Only a single value of dark current and light current were measured at each wavelength. The presence of 6 ADC levels between 87.2 and 88.2 %T is readily apparent. The signal has a noise component of ± 2 ADC levels. The slight downward slope of the lines is due to changes with wavelength from the value of 615 levels for 100 %T.

Figure 4b shows the same spectrum obtained by averaging 100 cycles of dark and light current at each wavelength. The noise has been greatly reduced by averaging. However a stair-step curve results with a step corresponding to each ADC level. Figure 4c is a repeat of Figure 4b except that random electronic noise was added to the signal from the photomultiplier before it was fed to the multiplexer input for conversion to digital. While some noise remains evident, the stair-step effect in Figure 4b has been largely eliminated.

### *Performance Checks*

For quantitative analysis, the primary factors in the performance of a spectrophotometer are linearity of response and reproductibility. At 600 nm and a slit width of 0.1 mm, repeated checks gave standard deviations of $8 \times 10^{-5}$ and $6 \times 10^{-4}$ absorbance units at zero and unity absorbance, respectively.

Linearity was tested with neutral density screens every 2 nm from 450 to 650 mm. The first screen averaged $0.4942 \pm 4.1 \times 10^{-3}$ absorbance units. The second screen averaged $1.4094 \pm 2.7 \times 10^{-3}$ absorbance units. The point by point addition of the two screens gave $1.9036 \pm 4.5 \times 10^{-3}$ absorbance units. When both screens were measured together, the result was $1.8999 \pm 10.8 \times 10^{-3}$ absorbance. It was concluded that there was no statistical difference between the addition of the two screens measured separately and the two screens measured together, and therefore the instrument response was quite linear. These screens are known to have small scattering effects that cause fluctuations with wavelength of the measured absorbance. Even so, the standard deviations given above are not large at the absorbance levels involved.

### Educational Opportunities

The students who participated in the elective course and in the automation of the Cary 16 have learned quite a bit about spectrophotometers, analog to digital conversions and programming. However, opportunities to learn have not ceased simply because the automated system is now working well. Unlike a turnkey type of system, the hardware and software are fully accessible to the student. The signals can be traced and examined at each point. Simple programs can still be written to operate the wavelength drive, or to sample the photocurrent on alternate cycles of the chopper, or whatever else the student dreams up. Additions to the system are still possible, such as stopped flow apparatus, automated sample changing, programmed temperature control, etc. In future years we expect to have students investigate the use of the programmed gain preamplifier as an alternate to the use of random noise. Also the use of a gated integrator can be compared to averaging discrete values.

In addition, the availability of the automated Cary 16 opens new possibilities for experiments in the required analytical laboratory. For example, it is now feasible to do an experiment to demonstrate the relationship between slit width, spectral band width, and resolution. Another experiment suitable for the required course would entail the drawing of Ringbom plots at a number of wavelengths and the calculation of photometric accuracy as a function of both concentration and wavelength. Still others can be based on the analytical applications of derivative spectra and log absorbance spectra.

### Summary

We believe that the approach used in the elective course discussed in this paper is a viable way to teach the concepts of data acquisition and instrument control by minicomputers. In addition, experiments can be developed that enhance the teaching of basic concepts of instrumental analysis.

Specific details of an experiment must depend on the particular instruments as well as the computer hardware and software. For this reason, specific details have not been given in this paper but will be supplied by the authors on request.

### Acknowledgments

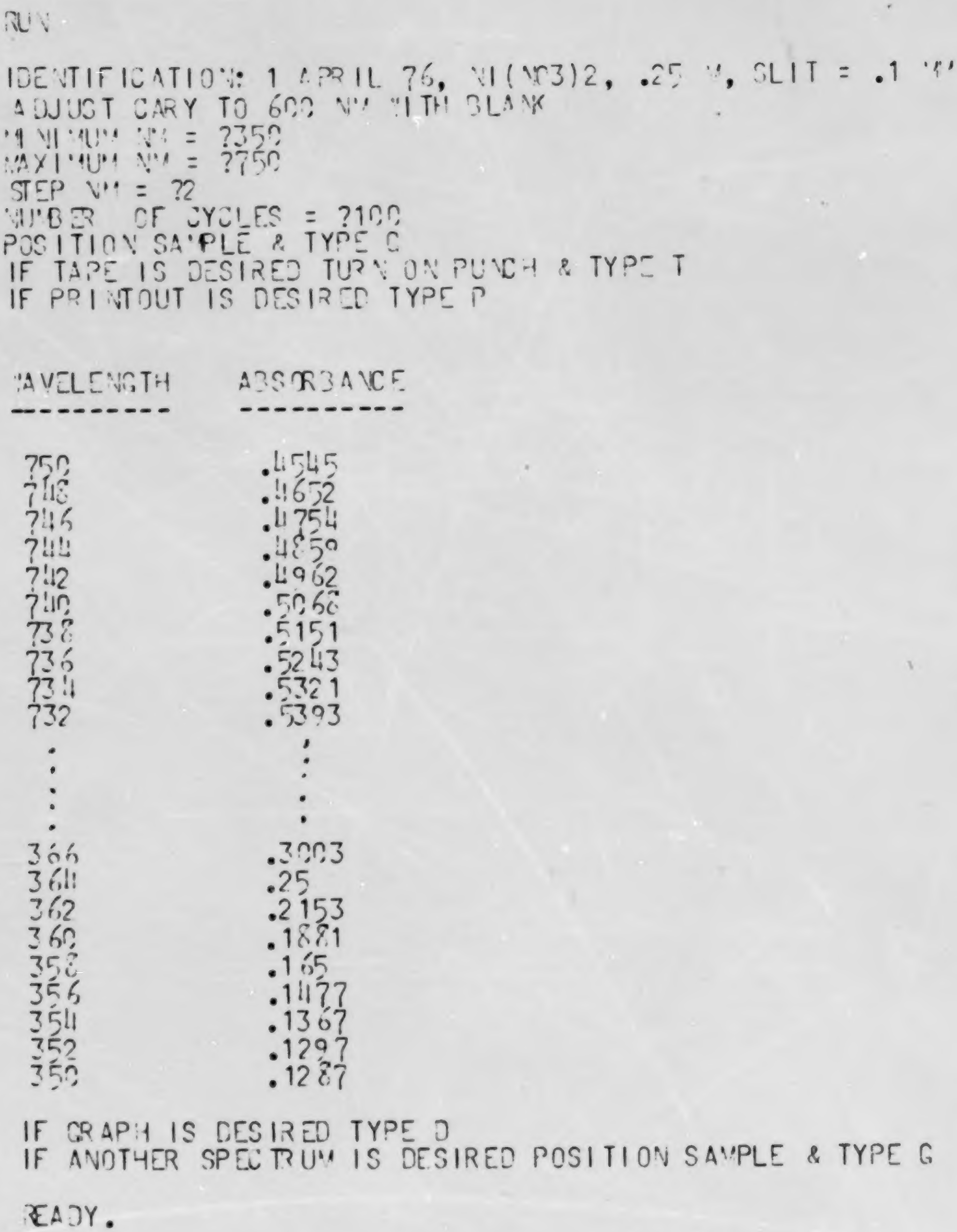
Portions of the computer system were purchased with a matching grant from the National Science Foundation Instructional Scientific Equipment Program. We are also grateful to the Digital Equipment Corporation for the loan of other portions of the computer systems.

### Note

Correspondence with regard to the contents of this paper should be directed to Professor Sturrock.

### References

1. EDUSYSTEM 10/5 (DEC-E8-U10A-PB). Digital Equipment Corporation, Maynard, Mass.
2. DAQUAN(DEC-LB-U80C-PB). Digital Equipment Corporation, Maynard, Mass.
3. BASIC R/T (DEC-LB-U70B-PB). Digital Equipment Corporation, Maynard, Mass.
4. Perone, S.P. and Eagleston, J.F., J. Chem. Ed. *48*, 317 (1971).
5. Flaschka, H. A., Private communication.
6. Horlick, G., Anal. Chem., *47*, 352 (1975).



```
RUN

IDENTIFICATION: 1 APRIL 76, NI(NO3)2, .25 M, SLIT = .1 MM
ADJUST CARY TO 600 NM WITH BLANK
MINIMUM NM = ?350
MAXIMUM NM = ?750
STEP NM = ?2
NUMBER  OF CYCLES = ?100
POSITION SAMPLE & TYPE G
IF TAPE IS DESIRED TURN ON PUNCH & TYPE T
IF PRINTOUT IS DESIRED TYPE P


WAVELENGTH     ABSORBANCE
----------     ----------

 750            .4545
 748            .4652
 746            .4754
 744            .4859
 742            .4962
 740            .5066
 738            .5151
 736            .5243
 734            .5321
 732            .5393
  .               .
  .               .
  .               .
  .               .
 366            .3003
 364            .25
 362            .2153
 360            .1881
 358            .165
 356            .1477
 354            .1367
 352            .1297
 350            .1287

 IF GRAPH IS DESIRED TYPE D
 IF ANOTHER SPECTRUM IS DESIRED POSITION SAMPLE & TYPE G

 READY.
```

Figure 1. Teletype dialogue for obtaining absorbance spectra with CARY XVI under computer control.

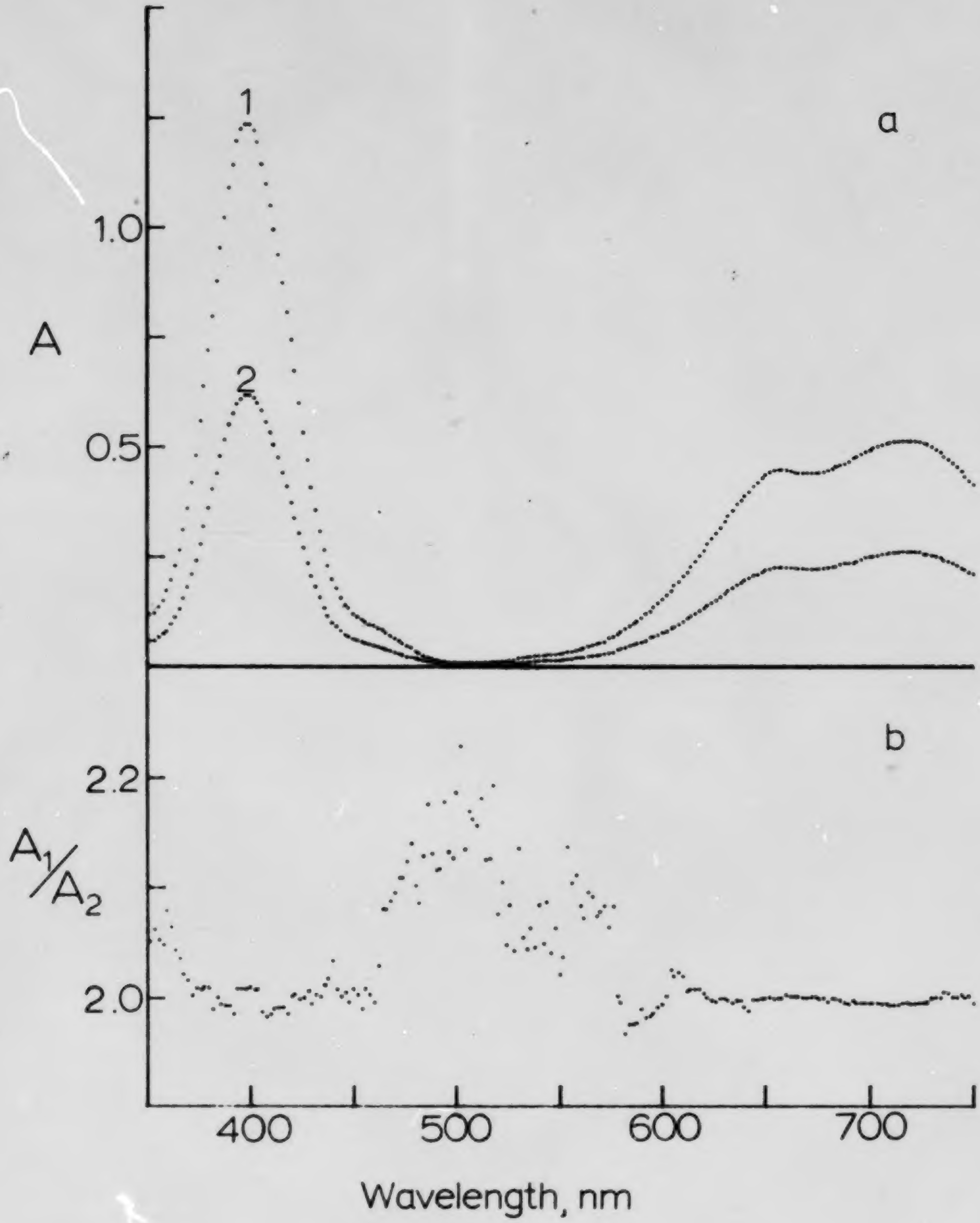


**Figure 2.** (a) Absorbance spectra of $Ni(NO_3)_2$. Curve 1, 0.25 M; curve 2, 0.125 M.

(b) Ratios of curve 1/curve 2 of Figure 2a.

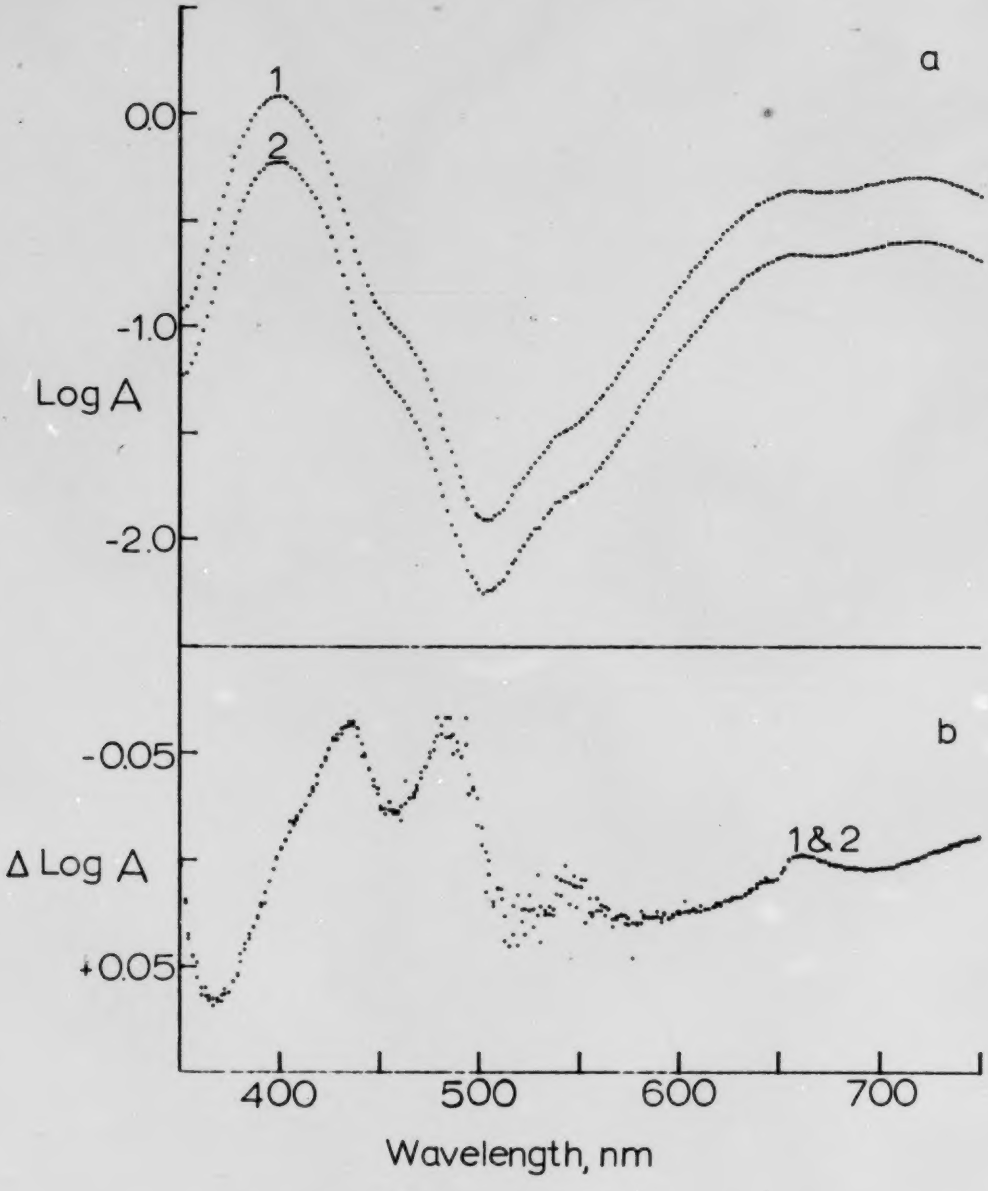


Figure 3. (a) Log absorbance spectra, same data as Figure 2a.

(b) Derivative log absorbance spectra derived from Figure 3a.

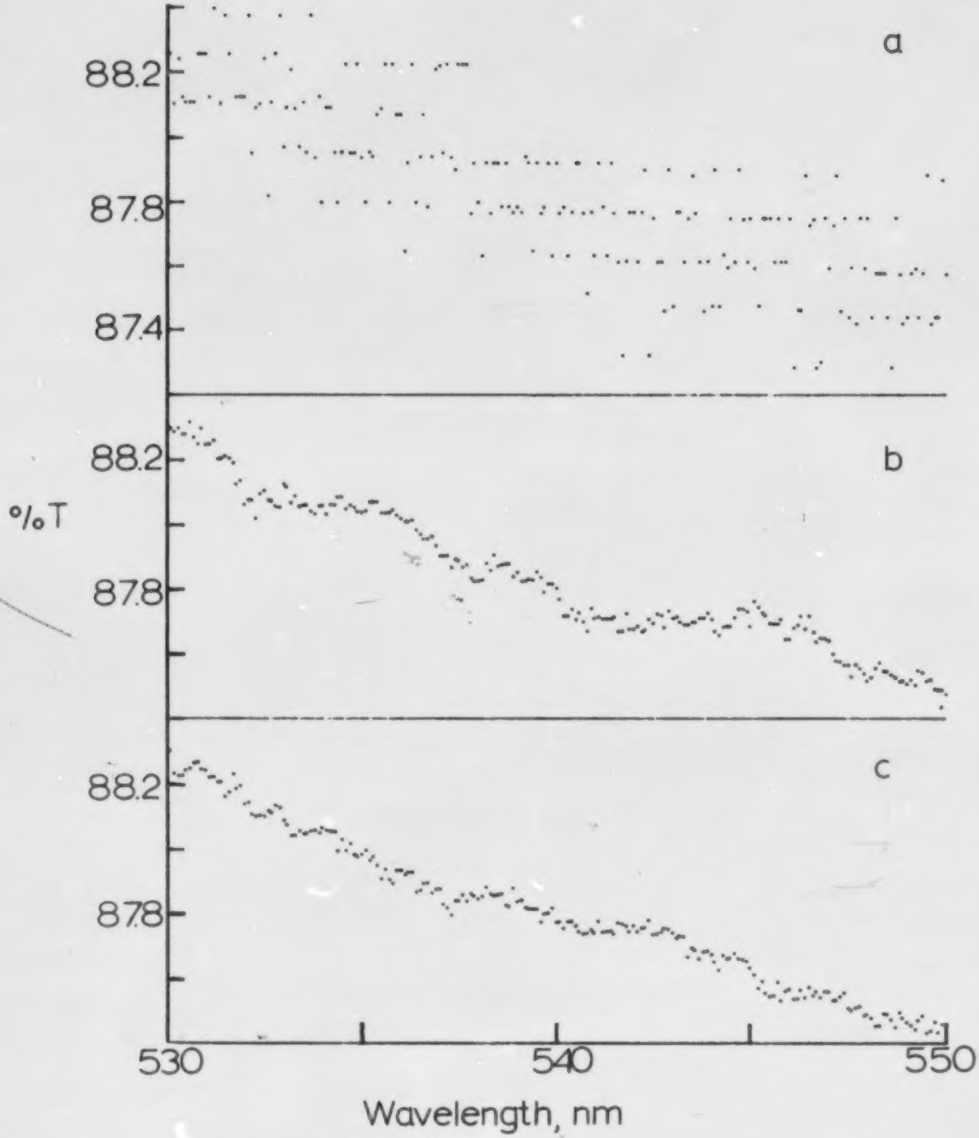


Figure 4. %T spectra of 0.5 M $Ni(NO_3)_2$

(a) One data point at each wavelength
(b) Average of 100 data points at each wavelength
(c) Same as b except with added random noise

# Using APL to Facilitate a Self-Paced Course in Cost and Optimization Engineering

**George N. DePuy**
*State University of New York at Binghamton*
*Binghamton, New York 13901*
*(607) 798-2631*

## Program Structure

A self-paced course in cost and optimization engineering was taught for the first time during the spring 1976 semester. Typical objectives of a self-paced course, such as allowing students to progress at different rates, were achieved. Also, the course was designed with the intent of having students learn more material for the same grade.

There are two features of this course which make it unique. First, students analyze "real" data which they collect from some situation of interest to them individually. Second, the computer is used extensively as a problem-solving tool in the course. Two purposes are served by these unique features. First, students gain experience in identifying a wide range of problems that lend themselves to computer analysis. Second, they develop an added appreciation for the sophistication possible with computer analysis.

For the initial offering, the course consisted of nine instructional units. These were devoted to the following topics: (1) Learning curves, (2) PERT, (3) CPM, (4) multiple correlation, (5) multiple regression, (6) chi-square test, (7) analysis of variance, (8) queueing theory, and (9) inventory theory (using Monte Carlo simulation).

Linear programming will be included as an instructional unit for the second offering of the course during the spring 1977 semester.

The course is taught at the senior, undergraduate level. There are two prerequisites. One is an introductory course in statistics which covers correlation and regression, but not chi-square and analysis of variance. APL is the computer language used in the course. Hence, a one-semester course in APL is a prerequisite.

A student's grade is determined by the number of instructional units completed. Eight units yields an A, six units a B, and so on. Instructional units are completed only when all requirements specified by the instructor have been satisfied. During the first offering of the course, students averaged about two submissions for each completion.

Students begin work on an instructional unit by attending a lecture on the topic and/or reading a prepared handout. The handouts contain: (1) a description of the theoretical basis for the topic being studied, (2) the requirements for completing the instructional unit, and (3) sample runs of the computer programs to be used.

Students individually identify situations which they want to study and collect data. The next step usually involves entering the data in an APL program, easily accessible through a computer terminal. After obtaining the output, the student is usually required to make a series of calculations. The calculations are one of two types. One is to duplicate some of the calculations done by the computer. The second, more common type, is to use the the results given to make some additional calculation not done by the program.

The student's final step is to prepare a written report on the instructional unit. The report must be concise, neat, satisfly all requirements on data collection and analysis, and demonstrate an understanding of the subject matter.

Seven of the nine instructional units involve computer programs. Units on multiple regression, chi-square, and analysis of variance use programs from the APL Public Library. The Public Library has programs on PERT and CPM, but these have not yet been incorporated into the course. When the instructional unit on linear programming is added to the course, an available Public Library program will be used. The unit on queueing theory uses a program written by a student under the direction of the author. The units on learning curves, multiple correlation, and inventory theory (using Monte Carlo simulation) use programs written by the author. Since the programs written by the author are not otherwise available, they will be made available to conference participants.

An evaluation was done at the end of the first offering of the course. Students felt they had done more work and learned more in this course than in other courses where they had received the same grade. The grade distribution turned out to be very typical for a senior level course (8 A, 11 B or B+, 6 C or C+, 6 D, 1 F, 3 W). Students indicated the computer work had a possitive effect on the course, and their attitude toward the course was very positive.

## Requirements for Completing Instructional Units

Following is a listing of the requirements for completing each of the nine instructional units. The theoretical basis for each unit is not discussed since no new contribution has been made here. However, references for each unit have been included. Space does not permit the listing of computer programs or displaying of sample outputs, but these will be discussed during the conference presentation.

### *Learning Curves*

1. Collect learning curve data on some product, process or activity of interest. Examples are: (1) product cost, (2) test time, (3) assembly time of a mechanical sub-assembly, and (4) assembly time of a jigsaw puzzle. Cost (or time) versus quantity information is required for a minimum of twelve units or lots.

2. If your data are for lots, use the LOT program to find the equivalent midpoints.

3. Use the LCS program to find the best one-slope, two-slope, and three-slope learning curves for the data.

4. Use the PLOT1 program to plot the best one-slope curve, PLOT2 to plot the best two-slope curve, and PLOT3 to plot the best three-slope curve.

5. Using the best one-slope curve as described in the output from the LCS program, calculate the total cost (or time) and the average cost (or time).

6. Choose a unit number beyond the range of the data and calculate the unit cost (or time) projected for that unit. Use the last section of the three, two, or one slope curve, depending on which has the best fit.

The LOT, LCS, PLOT1, PLOT2, and PLOT3 programs were written by the author. References for this unit are Abramowitz and Shattuck; Hein; Jelen; and Ostwald.

### *PERT*

1. Analyze a project consisting of at least twelve activities.
2. Draw the network diagram.
3. Find $t_e$ for each activity.
4. Find $T_E$ for each event.
5. Identify the critical path.
6. Choose a $T_L$ for the project and find the slack for each event.
7. Choose three values of $T_L$ and find the probability of completing the project on time for each.

The text by Levin and Kirkpatrick, in soft cover, is purchased by the students for this part of the course. Chapters one through seven in the text, instead of the prepared hand-out, are used to provide the theoretical basis for the subject. Available APL Public Library programs have not yet been incorporated into the course because they are event-oriented rather than activity oriented.

### *CPM*

1. Analyze a project consisting of at least eight activities.

2. Find the project cost for normal and crash times.

3. Choose a cost between the normal and crash costs and find the minimum time for this cost.

4. Choose a time between the normal and crash times and find the minimum cost for this time.

Chapter nine in Levin and Kirkpatrick provides the theoretical content for the unit.

### *Multiple Correlation*

1. Collect data on a situation of interest. A minimum of six independent variables and ten samples must be used. Examples of dependent variables are (1) salaries, (2) employee morale, (3) number of defects, (4) product quality, (5) power consumption, (6) church attendance, (7) grades, (8) rating of courses, (9) selling price of cars, (10) rating of airlines, and (11) number of vehicular trips made by families each week.

2. Form the data into a matrix—each column corresponding to a different variable. The dependent variable goes in the last column.

3. Use the CM program to produce the correlation matrix.

4. The last column in the correlation matrix gives the correlation between the dependent variable and each of the independent variables. Determine which of the coefficients are significant and at what level.

5. Use the MC2 program to find each of the two-variable multiple correlation coefficients.

6. Determine which combinations give significant increases over the highest single-variable correlation coefficient.

7. Use the MC3 program to find each of the three-variable multiple correlation coefficients.

8. Determine which combinations give significant increases over the highest two-variable correlation coefficients.

The CM program is from the APL Public Library. The MC2 and MC3 programs were written by the author. References for this unit are Ellis; Kerlinger; and Kerlinger and Pedhauzur.

### *Multiple Regression*

1. Using the data matrix which was compiled in the unit on multiple correlation, have the REG program operate on all independent variables simultaneously.

2. Using the t values given in the output, determine which variables are significantly related to the dependent variable.

3. Using the F value given in the output, determine if the overall regression is significant.

4. Do the analysis again with only significant variables. Using the t and F values in the output, again check the significance of the independent variables and the overall regression.

5. Write the multiple regression equation.

6. Use the REG program to find the slope and Y intercept for just one independent variable acting as if the others did not exist. Compare the results to the values in the multiple regression equation.

7. Use the REG program repeatedly to find the best four-variable, five-variable, and six-variable combinations. Combine these results with those from the previous unit. Prepare a table showing: (1) the best combination of variables for one-variable through six-variables, (2) the percentage of variability accounted for by each best combination, and (3) whether each combination gives a significant increase in percentage variability accounted for and, if so, at what level.

The REG program is from the APL Public Library. References for this unit are Kerlinger; Kerlinger and Pedhazur; Miller and Freund; Walpole and Myers; and Wonacott and Wonacott.

### *Chi-Square Test*

1. Collect sufficient data to form a matrix of at least three rows and three columns. For example, the rows might be first, second, and third shift. The columns might be the number of units produced with zero, one, and more than one defect.

2. Use the CHI program to find the calculated value of chi-square.

3. Compare this to table values to determine if the rows are significantly different and, if so, at what level.

4. Take the rows two at a time and use the CHI program to find the calculated values of chi-square. Compare these to table values to determine if the rows are significantly different and, if so, at what level.

5. Make the calculations by hand for one of the pairs in step four.

The CHI program is from the APL Public Library. References for this unit are Ellis; Miller and Freund; and Williams.

### *Analysis of Variance*

1. Collect data for a two-factor analysis of variance. It can be as simple as a two-by-two matrix. At least three data points are required for each cell. As an example, the rows might correspond to different machines and the columns might correspond to different operators. Each data point might be the number of units produced by a particular operator on a particular machine.

2. Use the TWOWAY program to analyze the data.

3. Use the output from the program to determine which main effects and interactions are significant and at what level.

4. Use the ANOVA1WAY program to do a one-way analysis of variance. At least three cells are required. If the matrix in step one had at least three rows (or columns), that data can be used by lumping all the columns (or rows) together. Determine if the differences are significant and, if so, at what level.

5. Do the calculations in step four by hand.

6. Use the ANOVA1WAY program to test for significant differences in each two-variable combination for the data in step four.

The TWOWAY and ANOVA1WAY programs are from the APL Public Library. References for this unit are Ellis; Kerlinger; Miller and Freund; Williams; and Wonacott and Wonacott.

### *Queueing Theory*

1. Obtain from the instructor individual values for the mean arrival rate and the mean service rate. Assume an infinite source population, unlimited queue length, poisson arrivals, exponential service time distribution, and first-come-first-served service discipline.

2. For the number of servers equal to four, calculate: (1) the probability of zero units in the system, (2) the expected queue length, (3) the expected waiting time in the queue, (4) the expected waiting time in the system, and (5) the expected line length.

3. Use the THEORY program to find the above values for five other vlaues for the number of servers.

4. Given the Cost of Serving = $5/hour and the Cost of Waiting = $10/hour, calculate the total cost when the number of servers is four.

5. Use the COST program to find the optimum number of servers.

6. Suppose the cost of waiting is not known. Perform a sensitivity analysis by trying a wide assortment of values, between zero and one-hundred dollars, for the cost of waiting. Prepare table showing the range of waiting costs for which different numbers of servers are optimum. Comment on the sensitivity.

The THEORY and COST programs were written by a student under the direction of the author. The reference for this unit is Hillier and Lieberman.

### *Inventory Theory (using Monte Carlo Simulation)*

1. Obtain from the instructor individual values for: (1) ordering cost, (2) carrying cost, (3) shortage cost for lost orders, (4) shortage cost for retained orders, (5) percentage of orders lost, (6) probability of lead times being one, two, three, or four time periods, and (7) the probability of demand being forty, fifty, sixty, seventy, or eighty units.

2. Use the INVSIM and RERUN programs to find the optimum combination of order quantity and reorder level. INVSIM must be used for the first interaction to enter the given values. RERUN is used on all succeeding interactions. Average inventory cost for the fifty-time period simulation is given on each interaction. Tabular and graphical output

showing values for each of the fifty time periods is optional. Ask for tabular and graphical output on the first and last interactions so the optimum situation can be compared with the starting point. Include a table showing the average inventory cost for each interaction.

3. Find the expected value for lead time and demand. Assume demand equals the amount used each time period. Using the equations given in the handout, calculate the optimum order quantity and reorder level.

4. Use the RERUN program with the calculated values for optimum order quantity and reorder level. Ask for tabular and graphical printout.

5. Compare the results obtained in steps two and four noting the differences in (1) optimum order quantity, (2) optimum reorder level, and (3) average inventory cost. Comment on why the differences occur.

The INVSIM and RERUN programs were written by the author. References for this unit are Jelen and Richards and Greenlaw.

## References

Abramowitz, J. G. and Shattuck, G. A., Jr., *The Learning Curve: A Technique for Planning, Measurement, and Control*, Report No. 31.101, Data Processing Group Headquarters, IBM, Harrison, N.Y., 1966.

Ellis, R. B. *Statistical Inference: Basic Concepts*. Prentice Hall, Englewood Cliffs, N.J., 1975.

Hein, L. W. *The Quantitative Approach to Managerial Decisions*. Prentice Hall, Englewood Cliffs, N.J., 1967.

Hillier, F. S. and Lieberman, G. J. *Introduction to Operations Research*. Holden-Day, San Francisco, 1967.

Jelen, F. C. (Ed.) *Cost and Optimization Engineering*. McGraw-Hill, New York, 1970.

Kerlinger, F. N. *Foundations of Behavioral Research* (2nd Ed.), Holt, Rinehart and Winston, New York, 1973.

Kerlinger, F. N. and Pedhazur, E. J. *Multiple Regression in Behavioral Research*, Holt, Rinehart and Winston, New York, 1973.

Levin, R. I. and Kirkpatrick, C. A. *Planning and Control with PERT/CPM*. McGraw-Hill, New York, 1966.

Ostwald, P. F. *Cost Estimating for Engineering and Management*. Prentice-Hall, Englewood Cliffs, N.J., 1974.

Miller, I. and Freund, J. E. *Probability and Statistics for Engineers*, Prentice-Hall, Englewood Cliffs, N.J., 1965.

Richards, M. D. and Greenlaw, P. S. *Management Decisions and Behavior*. (Revised Ed.), Richard D. Irwin, Homewood, Ill., 1972.

Walpole, R. E. and Myers, R. H. *Probability and Statistics for Engineers and Scientists*, John Wiley and Son, New York, 1972.

Williams, F. *Reasoning with Statistics: Simplified Examples in Communications Research*, Holt, Rinehart and Winston, N.Y., 1968.

Wonacott, T. H. and Wonacott, R. J. *Introductory Statistics* (2nd ed.) John Wiley and Sons, New York, 1972.

# TTP—A Computer Package for Teaching Transportation Systems Analysis

**Mark Abkowitz**[1]
*Transportation Systems Center, Code 231*
*Cambridge, Massachusetts 02139*
*(617) 494-2511*

**Steven R. Lerman**
*Transportation Systems Division*
*Massachusetts Institute of Technology*

## Introduction

In the academic forum, students of transportation engineering and planning are introduced to the theory behind analysis techniques, yet they rarely have an opportunity to employ them in a realistic transportation planning study. For example, students are introduced to the notion of transportation supply-demand equilibrium analysis (Manheim, 1975), but they may not have an intuitive grasp for how transportation supply is represented, how demand is represented, and how supply and demand interact to achieve an equilibrium solution. At a more micro level, the student may generally understand how routes, modes, etc., function, but rarely has more than a conceptual view of how these components are integrated into a complete, multi-modal transportation system. Moreover, the incorporation of these components into a computational package and the use of this package in the planning and design process are never clearly presented or available to the student. In summary there appear to be two levels of academic training which need bolstering, the internal understanding of the interaction of basic transportation relationships, and the external application of an analysis package in a transportation study.

## Research Development Effort and Results

The objective of the work described in this paper was to develop a computer-based transportation model system that could be understood and operated by students, and yet produce meaningful results for demonstrating the role of computer forecasts in the planning process. The economic constraint imposed by an entire class performing computer-based analyses as part of required curriculum was also considered.

Consideration was given to utilizing existing computer programs which were intended for non-pedagogical use. However, these programs were either too difficult or expensive to use, or did not utilize the basic theoretical concepts that are central to modern transportation systems analysis.

The principal product was a transportation analysis package dubbed the Transportation Teaching Package (TTP). TTP is a descriptive model system that predicts travel volumes in a multi-modal, network context. The package is structured for generalized application, where the user defines all the necessary inputs in representing the transportation problem under study. TTP is coded in FORTRAN IV for an IBM 360 or 370 computer.[2] The program is designed to be modular in structure to allow future users flexibility in adding future inputs, outputs, or model forms. Because the package is explicitly designed for analyzing problems of relatively small size, it proved possible to develop a sophisticated and representative program while maintaining small data requirements and low run costs.[3]

## TTP Capabilities

Perhaps the most important feature of the TTP package is the capability to represent a range of varied, multi-model transportation problems. The user defines a transportation problem by inputing a description of the structure of the transport network, the characteristics of the network, and the travel demand behavior of people served by the transport system.

The actual representation of a transportation network can be coded directly into TTP. There is no need to create so-called "dummy" links which meet the specific requirements of the algorithms employed to analyze the network. This allows the student to concentrate on coding his actual representation rather than worrying about software limitations. TTP explicitly allows many modes to operate from a single transportation analysis zone, a capability which realistically represents the effects of competing modes operating on the same network.

Another TTP feature is the ability to model the travel demand behavior of more than one socio-economic group in a given zone. Often, in a realistic transportation problem, a particular geographical area contains more than one socio-economic group.

An additional TTP representation feature concerns the ability to explicitly model transfers in transit routes. TTP allows the user to define transportation systems operating at particular headways, and is able to model transfers directly by tracing a traveler's path through the network and identifying where a traveler makes a transfer, if any.

Incorporated in the TTP framework to maintain a consistency with the current theory of travel behavior are several recent conceptual innovations in transportation system analysis. For example, recent studies have found that people are far more sensitive to out-of-vehicle time than in-vehicle time; on the average, most empirical studies have found travelers to be two to three times as sensitive to wait time or walk time than to in-vehicle time (Lave, 1969). Recognizing the implications of this finding, the TTP structure enables the user to distinguish between in-vehicle activities and out-of-vehicle activities which comprise a person's trip. In this manner, the proper sensitivity to travel time can be applied, and the actual characteristics of the traveler's trip are more accurately represented.

More generally, TTP incorporates the recent innovations in demand-related research. Behavioral demand modeling has been introduced in recent years, based on representation of the disaggregate unit (the individual traveler), and his or her decision of whether, when, where, and how to travel. This demand approach is based on random utility theory, in which each traveler is confronted with a set of feasible alternatives, one and only one of which is chosen (Manski, 1973; Ben-Akiva, 1973; Domencich and McFadden, 1975). TTP has adopted the most widely used choice model, multinomial logit, into the modeling framework. The user defines the utility of a travel mode for each socio-economic group as a linear function of in-vehicle time, out-of-vehicle time, and fare. In addition, the user may optionally include socio-economic characteristics such as income and auto ownership into the utility structure if he or she so desires. The multinomial logit model is based on the assumption that the specified utility functions are the expected values of a random variable with a known distribution (McFadden, 1973). The resulting model expresses the probability with which an alternative will be chosen by any given individual.

TTP possesses the desirable quality of being able to demonstrate, in an applied manner, current state-of-the-art theory in transportation systems analysis. In addition, understanding of the interaction of supply and demand components in forecasting, and the use of computer forecasts in a planning analysis are greatly enhanced through use of the package. These qualities are pursued further in following sections.

## TTP Format

To represent a transportation system in a TTP format, the analyst must define several aspects of the study area. Specifically, the network being considered, the definition of geographical zones, the definition of transport modes, and the selection of market segments must be determined. TTP models transportation supply through network representation, and demand using origin-destination flows and a multinomial logic choice model for modal split. Once the supply-demand relations are defined, an equilibrium solution is approximated, and travel impacts assessed on the basis of these solutions. This entire process is discussed in more detail in the following paragraphs.

Transportation supply is modeled in the TTP context using network representation and defining related characteristics of the network. Supply input includes link, line, and path data.

The physical structure of the network is modeled using link descriptions. Each segment in the network is represented by a link, and connected to other segments in the network at nodes. The configuration of the entire network is therefore represented by a series of links connected by nodes, with links intersecting at particular nodes (e.g., traffic intersection connecting two streets) as they would in an actual transportation system. Each link in a network may vary in its size, length, and ability to handle traffic, and these characteristics are input for each link in the network. All links are one-directional, and travel costs incurred on a specific link (e.g., expressway toll) can be input as a link characteristic as well.

The service characteristics of public transportation routes (rail, subway, bus) in the network are modeled through the specification of lines. A line is defined as a particular transportation route, operating over a series of links. Each line has a specific headway to define the frequency of operation. For example, a line may consist of a bus, operating over a series of streets with a headway of ten minutes. Lines are modeled in TTP by defining the links comprising the line, and the headway on the line.

The entire trip from an origin to a destination over the specified network is defined as a path.[4] In analyzing travel between two zones, there are several feasible paths which the travelers may consider for any given mode. The paths may vary by the mode being used, and also by the service provided by the links on that path. One feasible path between a suburb and the central business district (CBD) could be to walk from the origin, take a bus, transfer to rapid transit, and then walk to the destination. In the TTP input structure, paths are specified by origin, by destination, and by mode. Path input includes defining the links and possible lines associated with each path. Path costs (e.g., transit fare) are also input, when appropriate.

The demand for transportation is a function of the potential tripmaker's travel tastes, the level of service provided by the network, and the activities to which the tripmaker might travel. TTP models travel tastes in the user-specified parameters of the utility functions. The user defines a utility equation for each demand group-mode combination in the study area. Users may also specify that some modes do not serve some zones or socioeconomic groups. For example, transit service may not be universally available, and driving alone is not feasible for travelers without licenses.

Having defined travel tastes for the entire study area, the only other required demand input is the number of potential tripmakers traveling between each zone pair. This is often segmented by demand group to distinguish the various travel patterns made by each group. Identifying potential travel serves as a base from which the model can assign trips over the network.[5]

Given a fully defined network and set of demand relationships, TTP then applies an iterative solution technique to solve for equilibrium. The equilibrium solution occurs where the level of service provided at a particular volume equals the volume willing to travel at a particular level of service.

In TTP, the equilibrium solution is approximated by the use of incremental assignment (Ruiter, 1973). This technique begins by computing the level of service on the network assuming zero volumes, and selecting a minimum disutility travel path for each origin-destination-demand group-mode possibility. Once the minimum travel path for each available mode has been identified for an origin-destination pair for a particular socio-economic group, the expected percentage of travel on each mode is calculated using the logic model. Having determined the mode split under zero volume conditions, a small fraction of the expected travel volume is assigned to each mode and the links on the minimum path for that mode, and then the level of service is updated based on the new volumes assigned to the links in the network. This process is continued for several iterations, until virtually all travel volume in the study area has been assigned.

When assignment is completed, travel impacts are assessed. Outputs for all TTP runs include a printout of user input, and relevant impacts of network and zonal travel. The program also has a substantial error checking capability that provides appropriate, user-oriented diagnostic messages.

TTP outputs include the forecasted network, line, mode, and origin-destination impacts. Predicted volumes and travel times experienced on each link of the network are reported. Line volumes, mode volumes, and mode split are reported for consideration of how well a particular mode or line is able to attract ridership. Origin-destination outputs include average travel cost, in-vehicle travel time, out-of-vehicle travel time, total travel time, and travel volumes for each origin-destination pair. The user can specify disaggregation of origin-destination impacts by mode and socio-economic group, if appropriate.

These outputs from the TTP package provide the student with the basic information required to perform an analysis of transportation planning alternatives. A more recent version of TTP also provides the facility for users to define impacts as linear functions of passenger-miles and passenger-hours on an arbitrarily defined set of link-mode-socio-economic group combination. This feature provides a simple means for cost calculations, pollution emission forecasts, etc., within the program.

## Use of TTP in the Academic Environment

Students using TTP are supplied with two documents describing TTP and its application: (1) the TTP User's Manual, and (2) the Three Mode Case Study.$^6$

The TTP User's Manual contains an operator's guide for using the TTP system. A list of commands are described, formats are defined, and examples are given as to the use of each command. In addition, the logical position of each command in defining the network and demand characteristics is outlined. TTP prints error messages when inputs have been incorrectly specified; the TTP User's Manual contains a list of these error messages to ensure a rapid debugging process. A final component of the User's Manual describes a hypothetical study area and provides a step-by-step description of how this particular transportation problem is modeled and coded into the TTP program. In essence, the TTP User's Manual provides the student with an understanding of how to model the problem in TTP and code it into the TTP program.

The Three Mode Case Study demonstrates the role of TTP in a planning study. A hypothetical study area is constructed consisting of a suburb and CBD. Suburban residents can reach the CBD via three modes: rail, auto taken directly to the CBD, and auto taken to a rail station with rail used to reach the CBD. Network, population, and demand characteristics are given.

In the context of the case study, students are told that the regional planning agency is interested in improving the transportation system and considers a number of planning strategies. The costs of implementing and operating each strategy are given. TTP is used to forecast the expected travel flow as a result of each transportation alternative. The user benefits associated with improving the system are measured against the costs of implementation to determine the tradeoffs inherent in the potential alternatives.

Each student in the class is given about one week to review these documents before receiving the TTP assignment. The assignment is to consider the Three Mode Case Study and to develop other planning strategies which may result in a different mix of travel benefits and costs than the ones described in the study. Our experience indicates that students find it useful to work in pairs. This also reduces computation costs for the class. The students are expected to define transport alternatives and to model, code and run each strategy in TTP. They then use the TTP output to analyze the merit of each strategy. The student has unlimited flexibility in defining planning strategies. He can create a new mode (e.g., carpool), expand existing links, change fares or headways, or even impose exclusive lanes for use by specific modes. This type of exercise allows the student to become familiar with representing transportation problems in a supply-demand analysis, conducting the analysis, and then using the results as one would in a realistic planning study.

### Reaction to TTP

For the past two years, the assignment described in the previous section has been incorporated into the introductory transportation system analysis course offered by the Transportation Division, Department of Civil Engineering at the Massachusetts Institute of Technology. (The course is actually two separate catalogue listings, one for graduate students and one for undergraduates. In the two years TTP has been used, a total of about forty undergraduates and sixty-five graduates have taken the course.) In both classes, reaction to TTP and the planning assignment were extremely favorable. Many students expressed an interest in being able to conduct their own transportation study; several students became deeply "involved" in the assignment and proposed additional strategies and analyses well beyond the minimum requirements of the assignment. A few students later modified the program to expand it for use in another project.

Students were asked to provide an anonymous, written evaluation of the TTP exercise and generally described the assignment as a "welcomed relief" to the boredom of normal, more conventional assignments. Virtually all students agreed that the use of TTP provided an additional perspective of transportation systems analysis. There were only a handful of problems experienced with coding and running TTP, all confined to mis-specification of inputs on the part of the student.

Most of the limited criticism of the exercise was directed towards the structure of the written assignment students were asked to produce. One or two students expressed the opinion that while actually using TTP was a valuable experience, writing what was generally a long discussion of their results was not very useful. This may well result from the traditional lack of emphasis in the MIT curriculum on written, non-analytic assignments. It is the authors' belief that the emphasis placed on describing transportation options and their impacts in a comprehensible manner should, if anything, be increased.

Since the initial success of TTP at MIT, other universities have taken an interest in the package and its pedagogical capabilities. Northwestern University's Department of Civil Engineering and Transportation Center designed additional problem sets requiring use of the TTP package and included five of these problem sets for a transportation course that was offered in the fall of 1976. Again, reactions were extremely favorable. One of the assignments used at Northwestern was based on a more sophisticated case study developed by Abkowitz (1976). This study includes two suburban origins, a more sophisticated network structure and the division of the resident population into high and low income groups. This case study provides a wider range of insights into the equity issues inherent in transportation system design and highlights some methodological questions which do not arise in the Three Mode example.

## Conclusion

TTP was developed as a pedagogical package designed to enhance a student's understanding of the role and concepts underlying transportation systems analysis. Modeling and coding problems in TTP and using this package in a simulated transportation planning study were directed at bridging the gap between theoretical and applied concepts of transportation systems analysis.

Experience to date suggests that the use of TTP has been a successful addition to the academic environment. Students have been able to operate the program with little difficulty, and have reacted favorably to TTP's ability to assist in the understanding of transportation systems analysis and its role in transportation planning.

The potential uses of TTP in the academic environment are practically limitless. To date, TTP has been used to model transportation problems with several links, modes, and zones. The advantages of TTP in this context have been described in this paper. However, smaller problems can also be modeled (e.g., one link, two modes) to isolate the effects of a particular service policy, or to examine tradeoffs between different route configurations. Several forms of sensitivity analyses could also be conducted in this context. Because TTP is able to represent a variety of transportation problems at user-specified levels of detail, it has the potential to be a significant asset in developing a successful transportation analysis curriculum.

## Acknowledgments

The authors wish to acknowledge the contributions of Professor Marvin Manheim who provided significant guidance in the development of the early version of TTP, and Mr. Greig Harvey who assisted in the early design and programming. In addition, they wish to acknowledge the conceptual contribution of Mr. Earl Ruiter, a primary developer of the DODOTRANS system. This program was an antecedent of TTP, and the basic structure of the first TTP case study was derived from a case developed by Mr. Ruiter.

## Notes

1. Correspondence with regard to the contents of this paper should be directed to Professor Abkowitz.

2. Professor Frank Koppelman of Northwestern University has recently recoded a version of TTP for use on a CDC computer.

3. Average run costs on the MIT IBM 370/168 comptuer were less than two dollars.

4. Future versions of TTP are planned to obviate the need for user-specified path data.

5. The concept of fixed total origin-destination travel volumes is probably appropriate for work travel only. A version of TTP currently being developed by Greig Harvey will allow users to include models of frequency, destination, and mode choice. See Adler (1975) or Charles River Associates (1972) for examples of such models.

6. This case study is based on one developed by Mr. Earl Ruiter while employed at MIT.

## References

Abkowitz, Mark. *Design, Development and Application of a Sketch Planning Package in Transportation Systems Analysis.* S. M. Thesis, Department of Civil Engineering, MIT, Cambridge, Mass., 1976.

Adler, T. J. *A Joint Disaggregate Model of Non-Work Urban Passenger Travel Demand.* S. M. Thesis, Department of Civil Engineering, MIT, Cambridge, Mass., 1975.

Ben-Akiva, Moshe. "Structure of Passenger Travel Demand Models." *Transportation Research Record 526.* Transportation Research Board, Washington, D.C., 1974.

Charles River Associates, Inc. *A Disaggregated Behavioral Model of Urban Travel Demand.* Federal Highway Administration, U.S. Department of Transportation, 1972.

Domencich, T. and McFadden, D. *Urban Travel Demand: A Behavioral Analysis.* North-Holland, Amsterdam, 1975.

Lave, C. H. "A Behavioral Approach to Modal Split Forecasting." *Transportation Research.* Vol. 3, No. 4, December 1969.

Manski, C. F. *The Analysis of Qualitative Choice.* Ph.D. Dissertation, Dept. of Economics, MIT, Cambridge, Mass., June 1973.

Manheim, Marvin. *Fundamentals of Transportation System Analysis.* Manuscript of text, 1975.

Manheim, Marvin L. and Earl R. Ruiter. "DODOTRANS I: A Decision-Oriented Computer Language for Analysis of Multimode Transportation Systems." *Highway Research Record 314.* Highway Research Board, Washington, D.C., 1970.

McFadden, Daniel. "Conditional Logit Analysis of Qualitative Choice Behavior" in Zarembka, Paul, ed. *Frontiers in Econometrics.* Academic Press, New York, 1974.

Ruiter, Earl. "The Prediction of Network Equilibrium—The State of the Art," paper presented at the International Conference on Transportation Research, Transportation Research Forum, Bruges, Belgium, 1973.

# Teaching Basic Computer Simulation Techniques to Undergraduate Biological Science Majors

**J. D. Spain**
*Biological Sciences Department*
*Michigan Technological University*
*Houghton, Michigan 49931*
*(906) 487-2029*

## Abstract

This paper describes our experience with the use of the programmable calculator or microcomputer for the instruction of basic simulation techniques to undergraduate biological science majors. This course has operated for five years with an average of forty students per year. Students working individually on an *ad lib* basis are capable of programming about twenty simulations which deal with a variety of subjects ranging from population dynamics and genetics to physiology and biochemistry. As a result of the course, the students become aware of a mathematical basis to biology which cannot be fully appreciated in any other way. The techniques employed in the course have worked very effectively with the microcomputer, but should be applicable to almost any computer hardware available.

## Background

The use of the programmable calculator of microcomputer for developing and demonstrating fundamental biological simulations was first described at the Conference on Computers in Undergraduate Curricula held at Iowa City in June 1970.[1] This paper showed how some of the classical models and simulations of biology could be implemented on a very primitive microcomputer, the Olivetti P-101. Some preliminary results on the use of these simulations in undergraduate coursework were described. By 1972, a formal course in biological simulation techniques had been added to the curriculum at Michigan Technological University, and was considered to be sufficiently unique and futuristic to be described in the *A.I.B.S. Education Division News.*[2] Since that time, the course has been modified in a variety of ways. The objective of this paper is to present some of the observations and conclusions which have resulted from five years experience with this course.

## Hardware

Shortly after the course was instituted, the Olivetti P-602 microcomputer became available, and it is the experience with this equipment which we will describe below. For those that are not familiar with the P-602, it is a desk-top machine which employs up to thirty-six storage registers for holding constants or variables. Programs of up to 384 instructions may be stored in the basic machine (up to 128 without imposing on the data storage registers). All data input, program instructions, and data output are printed at a rate of thirty characters per second on paper tape. This capability is essential for simulations, since they usually involve printing out a series of time dependent data. Compiled programs may be stored on magnetic program cards for future use. Those having experience with the programmable calculator will recognize that there is a variety of desk-top machines which meets these specifications.

## Basic Philosophy of the Course

In our early experience working with simulations, it became apparent that although prepared computer simulations offered tremendous possibilities for teaching basic principles of biology, an even more effective use of computers involved having the students program their own simulations. With prepared simulations, students often take a "so what" attitude which is not apparent when the student is actively involved in working out the simulation for him or herself. The advantage of the programmable calculator for such a use is that it allows the instructor to place the primary emphasis of a simulation course on the mathematical concepts of simulation rather than the problems of programming. Students are also spared the frustrations that are often associated with input to and out-put from major computer facilities.

### General Description of the Course

On successful completion of the biological simulation techniques course (BL-443) the student receives three quarter hours credit. It is expected that the student will have put in about ninety hours of work. Twenty hours will be spent in lecture or recitation, and the remainder will be spent in development and testing of programs. Grades are based primarily on the number of successfully completed simulations and models. By allowing the students free access to the lab, the four available machines more than adequately serve the forty students enrolled. Students spend approximately two hours per week with a machine. Additional time is spent in the preparation of programs. The course is taught on the senior level, although some juniors and graduate students also elect to take the course. Prerequisites for the course include one year of general biology and one quarter each of calculus and statistics.

The basic objective of the course is to give the student direct experience with the development and testing of a representative sample of about twenty different simulations. Students are asked to select these from a list of about thirty-five which deal with a range of fields in the biological sciences. Since most of the background information is provided in the instructional manual, students are encouraged to work at their own speed on any topic in the book even though the material has not yet been discussed in the lecture section.[3] The course outline provided below is based upon the one which was employed during the winter of 1976.

### Course Outline

*Week One.* Basic programming techniques for the P-602. (Laboratory emphasizes the programming of simple algebraic equations, looping and data storage.)

*Week Two.* Programming conditional jumps, counting, and use of mathematical subroutines. (In laboratory students prepare programs involving conditional jumps and counting.)

*Week Three.* The concepts of modeling and simulation. The use of computer flow charting in the development of models. (In laboratory students begin programming simple models.)

*Week Four.* Model equations based upon theoretical considerations. Empirical equations and curve-fitting. (In laboratory students work with prepared programs to see how models are distinguished from simulations, and how to employ simple curve fitting techniques.)

*Week Five.* The deterministic approach to simulation. Simple techniques for the numerical integration of differential equations. (In laboratory students work with simulations involving differential equations for growth, population dynamics, diffusion and chemical kinetics.)

*Week Six.* Ecological simulations dealing with population dynamics, life cycles, age classes, and energy flow. (In laboratory students work with simulations for competition, epidemic, predator-prey interaction, and energy flow.)

*Week Seven.* Simulations based on Hardy-Weinberg principles. Demography and world models. (In laboratory students work with Hardy-Weinberg simulation, balanced genetic load, and demographic simulation.)

*Week Eight.* Simulation of physiological systems. Compartment models, heart valve, feed-back control and temperature regulation. (Students prepare simulations of extracellular fluid space determination, enzyme kinetics, iodine compartment model, and temperature control model.)

*Week Nine.* Random number generators. Statistical analysis of computer generated random numbers. (Students generate a list of about 500 random numbers and perform various tests on them.)

*Week Ten.* Stochastic Simulations. Use of random number generators to develop the Monte Carlo type of simulation. Cueing. (In laboratory students develop simulations of the stochastic type. Dice throwing, random walk, monohybrid and dihybrid genetic crosses.)

## Student Evaluation

Students feel that the amount of work required is about right for the three credits obtained. They prefer the grading system which is based strictly upon the number of correct programs which are turned in. They like being able to work at their own pace. Students do not like working with prepared simulations. They much prefer working on simulations which they develop themselves. Many have asked to be given more opportunity to work on simulations in their specific field of interest. Many students remarked that the course gave them a whole new insight into certain aspects of biology. Essentially all students find the course to be a valuable and stimulating experience, and no student has complained that it is beyond his mathematical capability. It should be noted however, that this is an elective course, and presumably students with weak mathematical background do not enroll.

One year we attempted to have each student develop a simulation in his specific field of interest. This did not work out very well because students still need the material simplified and pre-digested before they can use it effectively at this level. Moreover, not all fields lend themselves to simulation. We do provide plenty of extra exercises in the manual so that the student may choose from a broad range of subjects. It should be noted that not all of the students complete the course. About 30 percent simply do not have the self discipline necessary to complete a course that is set up on this grading basis.

## Conclusion

This has been a very gratifying course from the instructor's standpoint. Students learn to use the microcomputer quite rapidly in part because it is not quite as awesome as the larger computer. They are enthusiastic about writing programs and developing simulations. There is no difficulty in stimulating class discussion as there are always questions and comments arising from the laboratory portion of the course. The amount of work accomplished is considerably more than is normally accomplished in a computer course. We believe this is the result of using the microcomputer, over which we had full control and minimal machine breakdown. It is clear that the programmable calculator offers great potential for the instruction of students in the theory and techniques of developing simulations.

## References

1. Spain, J. D., 1970. Mathematical Modeling of Biological Processes. *Proc. of the Conference on Computers in Undergraduate Curricula.* Univ. of Iowa. 1: 61-67.
2. Spain, J. D., 1972. Teaching Basic Biological Simulation Techniques with the Programmable Calculator. *A.I.B.S. Education Division News* 1: 5-7.
3. Spain, J. D. 1972. "Biological Simulation Techniques." Michigan Tech Press. Houghton, pp. 176.

# Enriched Understanding of Simulation by Usage and Dissection of PASS IV

**Harvey Axlerod**
*Academic Computing Services*
*4250 Ridge Lea Road*
*SUNY at Buffalo*
*Amherst, New York 14226*
*(716) 831-1761*

**Warren Thomas**
*Department of Industrial Engineering*
*SUNY at Buffalo*

## Introduction

Traditionally the teaching of discrete simulations has utilized the computer as a high speed tool via languages such as SIMSCRIPT and GPSS. This paper describes a more complete approach used by the Department of Industrial Engineering at SUNY/Buffalo. The PASS IV system is used not only as a tool, but also as a case study of the theoretical and design problems behind implementing such a language. The students first use PASS IV as a black box; later, they open the box for critical examination. This paper addresses the method in two parts: first, a brief overview of PASS IV, and secondly a description and evaluation of how this method works in a classroom environment.

## PASS IV

In 1965, Gordon Lilly wrote PASS as a simulation tool, implemented on an IBM 1620. The intent was to provide engineers with a small-core, FORTRAN-oriented system similar to SIMSCRIPT. It was then used primarily to simulate steel-mill operations.

In 1971, Harvey Axlerod and Lilly[2] redesigned the system, with the additional objectives of coding ease, extended error-detection, powerful debug facilities, and improved documentation. PASS IV was implemented on the SUNY/Buffalo CDC 6400 with the intent of providing the academic community with an easy-to-learn simulation language (CDC had only a poor version of SIMSCRIPT, and no GPSS) Except for some slight modifications necessitated by changes in the operating system (from SCOPE to KRONOS to NOS) and hardware (from CDC 6400 to CDC CYBER 173), the structure and coding have remained unchanged.

Because PASS is simply a set of FORTRAN subroutines, the student easily adds his own main program and subroutines, needing only to understand the rules of FORTRAN programming. The structure of PASS is such that a student is forced to define explicitly the primary events, the occurrence of which will cause a change in the state of the system. Once a student has defined the primary events, the calendar provides an efficient means for future events in chronological order, while subroutine SUPER enables the student to have the next event removed from the calendar and made ready for further execution.

Twenty-three lists are available for the student to file and retrieve from various queues. PASS takes care of all file maintenance tasks, an onerous burden were the modeling done

263

directly in FORTRAN. ITEM serves as a working register which the user is allowed to access and modify. The interface between ITEM and an entity in memory makes the logical flow of execution clear to the user. Each temporary entity contains sixteen attributes (assigned when in ITEM), permitting the easy assignment of characteristics to these entities as they flow through the system.

These features enable the student to develop his or her model more quickly than would otherwise be possible.

Self-protection of essential system data (queues, user-registers, various flags, and counters) is accomplished by storage in two distinct areas. COMMON/PASS/ provides the user with access to those variables he can define and/or interrogate, e.g., size of queues, structure of queues, current (simulated) time, and current event index. COMMON/QUIET/ is used only among the PASS IV subroutines, and is not declared by the user. It stores internal system tags which the user must never access, e.g., list boundary definitions, garbage collections, and queue linking. This distinction tends to make the data base fairly user-proof.

Queues are allocated dynamically, and are maintained as double-linked lists. Garbage is collected automatically into a special forward-linked list. Queue size is not limited individually but rather is constrained collectively by the total memory allocation. Empty (or unused) queues cost nothing.

The concept of a well-defined simulation data-base was coupled with the definition of the minimal queue operators needed. Thus, the user could custom code his own special queuing situations into "queue-macros" by use of the primitives provided. This lack of built-in clutter makes the system easy to comprehend.

Debugging facilities are few but powerful: snapshots, tracing, and autopsy. SHOWQ allows the user to snapshot any queue(s) at any time(s) desired. MONIT (time) initiates the tracing of all events and their effects on the system, including calendar schedules and diary records. Execution of a PASS-defined error will automatically initiate a call to DUMPAL, which terminates the run and autopsies the entire system. The user may define his own errors, and use DUMPAL to autopsy the system for him.

Thorough documentation was given high priority during implementation. Internally, the code is written mnemonically and generously peppered with comment cards. The user's manual features complete code/listings of the entire system. From the language security viewpoint, this was our worst mistake, as several modified, often degraded or non-working versions are floating around, with credit to the original authors. However, viewed pedagogically, this proved of immense benefit, as will be seen later.

## Pedagogical Applications in Engineering

PASS IV has found regular use in industrial engineering 475, *Experimental Methods*, which presents senior-majors with the methodologies of digital simulation modeling. Students approaching the learning of digital simulation techniques are expected to have a background in FORTRAN IV programming equivalent to that afforded by an introductory level computer science course. Most students also have some notions of mathematical modeling from earlier operations research courses. These notions are not absolutely necessary to learn simulation modeling; however, they enable the student to put simulation into perspective vis-a-vis other modeling approaches.

Twelve years of offering simulation courses to its students has convinced the faculty of the Department of Industrial Engineering that for effective learning to take place:

1. Students must be able to actually run their models.
2. A simulation language must be available to facilitate model building (FORTRAN alone is too tedious).
3. The language should be sufficiently simple as to permit learning of essentials in a brief time so that actual usage begins quickly.
4. The language should either possess the general features of more powerful languages

available in the field or clearly motivate the need for them.

5. The students should feel they can control the language rather than be controlled by it.

In earlier years GPSS was used. While easy to learn, it forces the student to concentrate on the details of GPSS without the opportunity to understand more fundamental notions common to all languages. In 1971, the search for a better instructional language led quickly to the use of PASS IV. Having the PASS authors on campus available for consultation provided a higher level of consulting than could be offered by the instructor. The capability for users to define their own error conditions in conjunction with DUMPAL expedited program development. A typical first problem assigned to students is the Toll Gate Problem, included as Appendix A.

Early in our usage of PASS, however, certain deficiences were noted. The first reaction was to degrade the utility of the language. However, it soon became obvious that these deficiencies provided excellent teaching opportunities. While forcing us to conclude that PASS is of limited usefulness in a production environment, these caused us to recognize its excellent pedagogical characteristics: students were prepared to learn quickly any other simulation languages they might encounter once they have left the university. Discussed below are three "advantageous" deficiencies:

1. There is no report generator capability. Thus, in order to obtain such common output statistics as facility utilization, average and maximum queue length, average transmit time and distributions of any of these, the student is forced to define variables, initialize them, increment as appropriate, and at run-end cause final calculations to be made and properly displayed. This consideration led to the definition of a user written STAT routine to collect time-averaged (system) statistics, and a REPORT routine to cause printing of the final results. This experience develops in each student an appreciation of report generators in other languages. More importantly, it develops a critical perceptability of what those generators are doing (or should do).

2. The statistical routines provided by PASS for generating random numbers and variates from the exponential, normal, and empirical distributions were found to be poorly conceived. The student is forced, as a class assignment, to define his or her own random variate generators. However, because the complete PASS program is listed in the manual, the student has ready access to all its details. Thus the instructor is able to use these poorly designed generators to develop in his students the skill to critically analyze random variate generators. This has proven to be extremely beneficial. Our students develop the ability to analyze this important part of a language. Often they find themselves, when in full-time employment with a company, as the only ones around who do understand the methods and limitations of various statistical generators.

3. PASS provides exactly sixteen attributes for every temporary entity. For classroom problems this number is usually more than enough, thus incurring inefficiences in memory space usage and execution time. For production problems in industry, sixteen attributes are frequently too few, thus preventing the use of PASS. This illustrates to the student the desirability for being able to vary the number of attributes assigned to an entity.

These apparent limitations in each case have provided excellent opportunities for the instructor to expand students' understanding of simulation languages.

Near the end of the course the students are given a lecture (by Mr. Axlerod) on the internal workings of PASS. While not needed to use PASS, these details enable the students to gain a more comprehensive understanding of simulation languages to prepare them for learning other specific languages—GPSS, SIMSCRIPT, GASP, SIMULA, CSL, etc. Coming as they do after the student has a good grasp of PASS, the lectures are usually well understood by the students. Their prior experience using PASS motivates their understanding of these advanced concepts.

## Appendix A

## Toll Gate Problem

Automobiles must stop to pay $.15 toll at the Black Rock toll barrier. There are three gates open: Lane A is an attended booth for customers who do *not* have correct change. Lanes B and C are automatic booths for customers who have the correct change.

There is a queue associated with each gate. Fifteen percent of the arrivals do not have the correct change and, therefore, immediately enter the Lane A queue. The 85 percent of the customers who have the exact change will enter queue B or C, whichever is shorter. If both queues are equal in length, he will select B or C randomly with equal probability.

The time between arrivals of customers is exponentially distributed with a mean of 3.0 seconds. The service time at gate A is uniformly distributed 10 to 30 seconds. Gates B and C have a constant service time of 6 seconds.

After 500 customers have been served, what is:

1. Percentage utilization of each booth?
2. Maximum and average length of each queue?
3. Average time a customer waits in each queue?

## Note

All correspondence regarding the contents of this paper should be directed to Professor Axlerod.

## References

1. Lilly, Gordon, *Parallel Activity Simulation System Manual*, Jones and Laughlin Steel, Pittsburgh, 1965.
2. Axlerod, Harvey and Lilly, Gordon, *PASS IV User's Manual*, State University of New York at Buffalo, 1971.

# Spatial Marketing Simulation: A New Package

**Robert S. Ellinger**
*Department of Geography*
*The University of Iowa*
*Iowa City, Iowa 52242*
*(319) 353-3131*

---

## Introduction

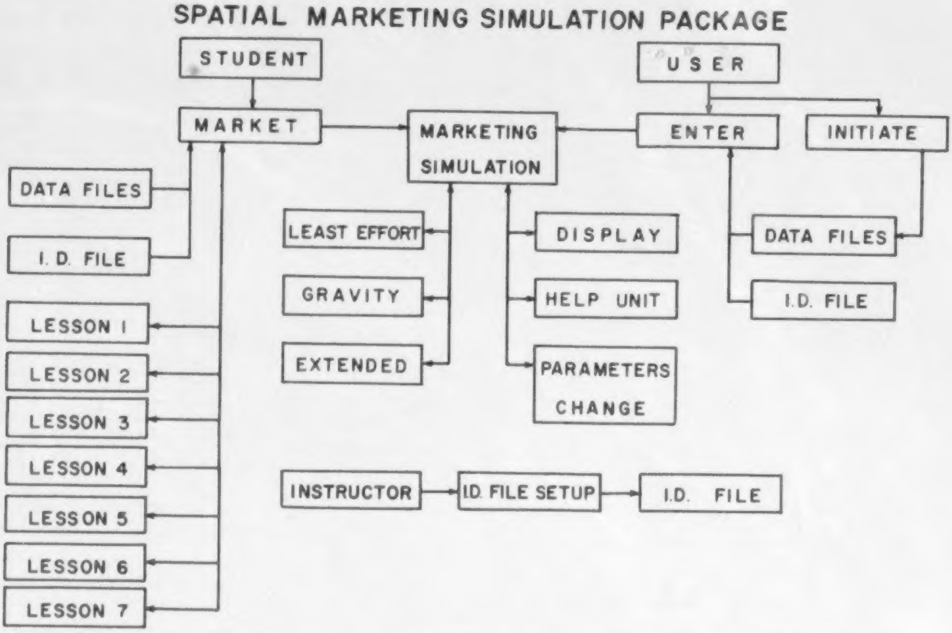
The development of the field of economic spatial analysis in geography in the past twenty years has produced a number of concepts of spatial processes or economic processes with spatial consequences. While traditional teaching methods are nominally adequate for the individual concepts, the undergraduate has a difficult time integrating the processes and perceiving the way the individual processes work within the whole, since traditional teaching methods are able to illustrate dynamic processes using only semi-dynamic techniques. The effective use of graphics CRT coupled with a sophisticated simulation would seem to provide a sound and efficient solution to these problems. This was the main purpose for development of the interactive spatial marketing simulation computer-assisted instructional package.

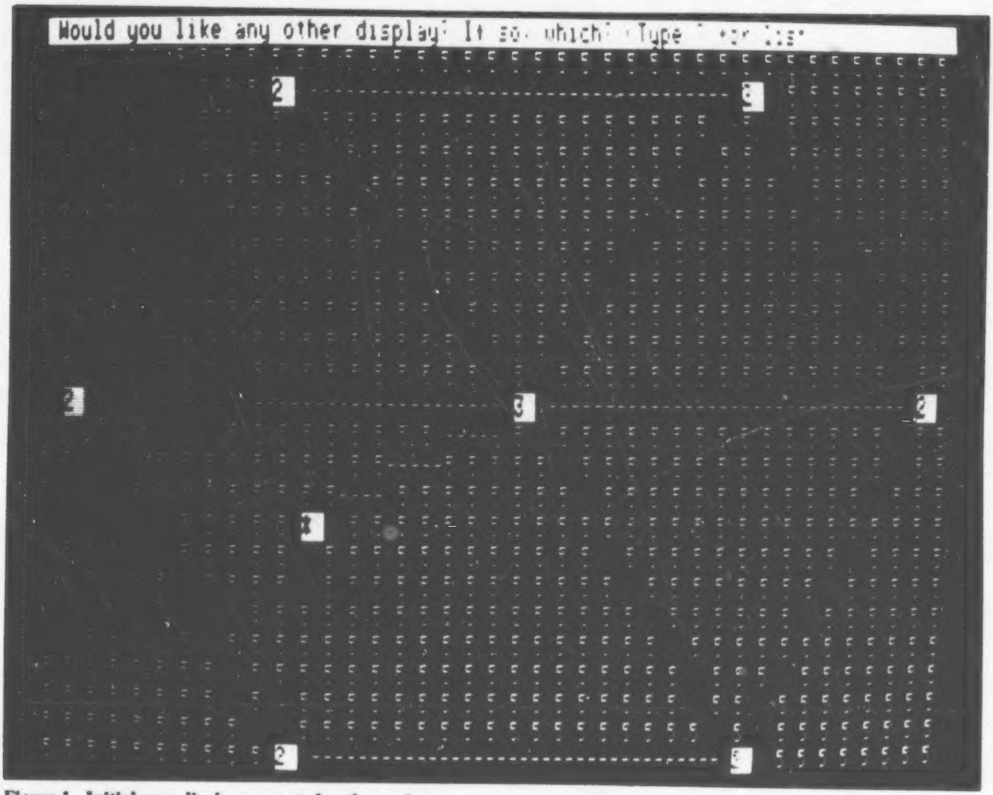
## The Simulation Model

The core of this package (as illustrated in the flowchart) is a highly flexible spatial marketing simulation model that gives the student the ability to replicate the location of markets, the transportation system, the population distribution, and various economic and cultural parameters of a region, within the limitation of the hardware available. The simulation has a number of possible displays including the location of firms by goods or by order, markets by order, the market regions for particular goods, and lists of variables and parameters. All variables and parameters pertaining to anything shown in any of the displays are stored in files and can be altered as desired. The simulation can be directed to use any user's set of files, thus enabling the student to apply the simulation to the student's own research in spatial marketing.

Added to the flexibility of the display, and equally important, is the ability of the simulation to call on any of three consumer models, at the student's discretion. These models are the least effort model, used in much classical spatial marketing literature; Huff's gravity model;[1] and a complex consumer model, developed for the simulation model, incorporating many concepts—action space, multiple-good tripping, shopping versus convenience-good behavior, and socio-economic characteristics. This gives the student using the simulation an immediate choice of consumer behavior models of graduating complexity to apply to a problem. For finding rudimentary market areas, the least effort model is sufficient. For researching the ways that cultural bias affects the locational stability of markets, the complex consumer model is needed.

Figures 1 through 4 illustrate a typical iteration in a model run. The initial configuration is shown in Figure 1. The student then reconfigures the display to suit his needs (Figures 2 and 3), and prints out the market areas (Figure 4). The student can then change the display or any location of firms or goods parameters, and rerun the model.





```
Would you like any other display? If so, which? (Type ? for list)
```

Figure 1. Initial map display presented to the student.

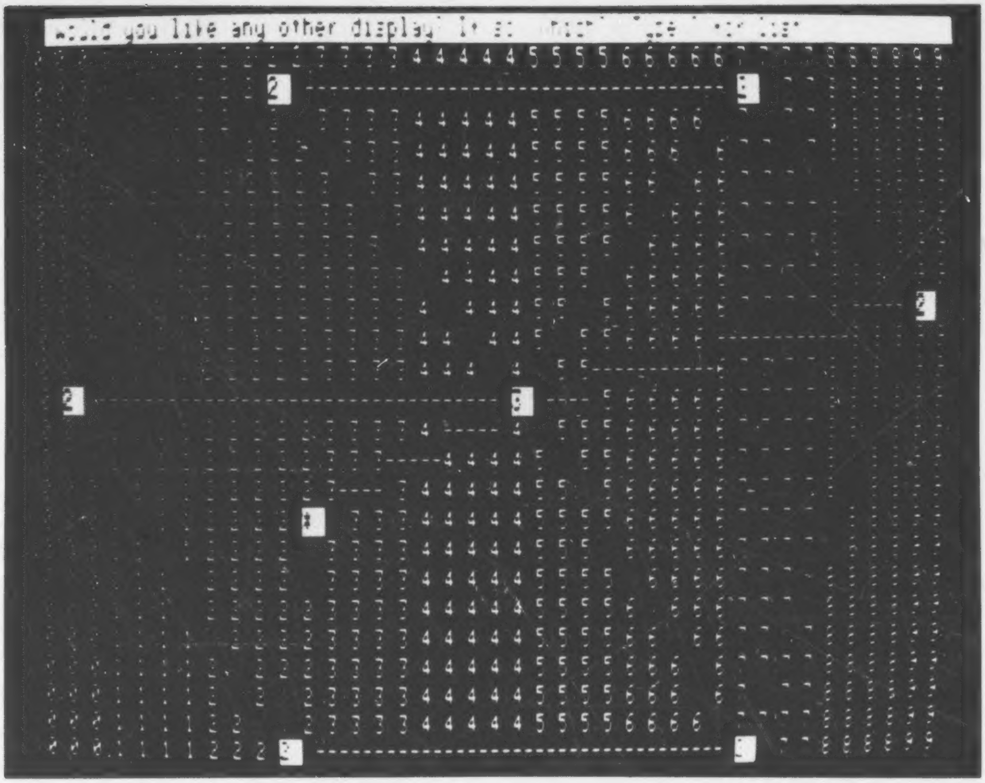


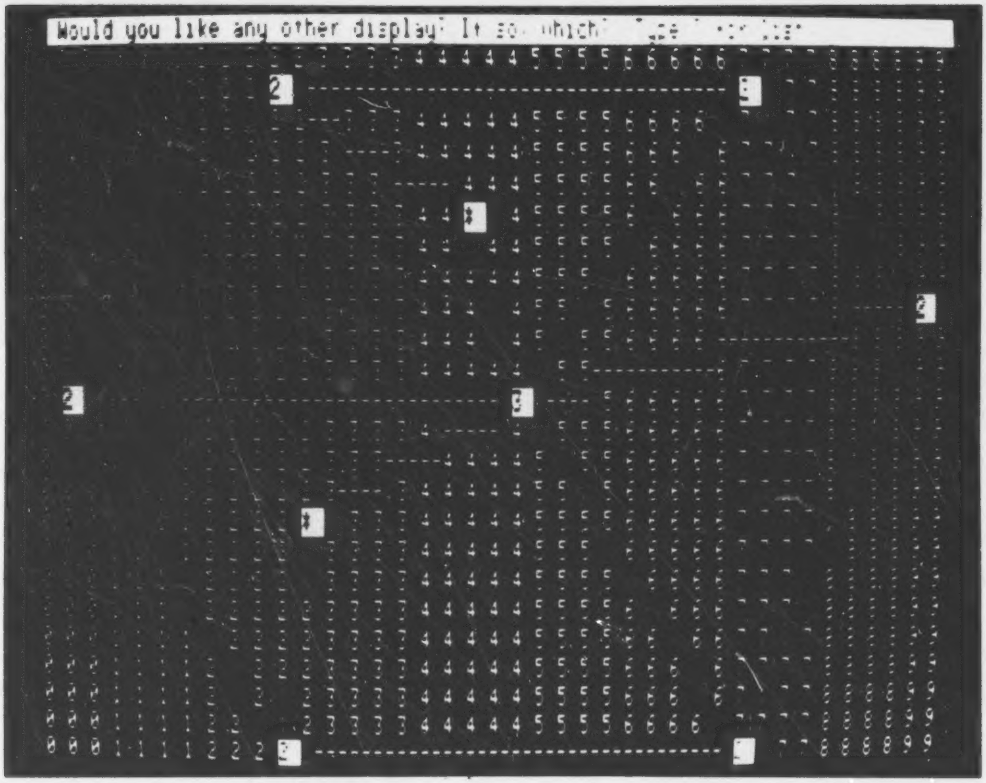
Figure 2. The student has changed the population density from even to sloping.

In addition to these models, a student who has had some training in programming can construct his own consumer model to fit some particular need, or he can use the model to display the results of some location-allocation algorithm. The simulation is equipped with a program that can give the student a copy of the major displays, which the student can use for later analysis.

While the hardcopy program is designed to work on any terminal, the simulation model itself must be used on a CRT equipped with cursor-addressing capability, reverse video, and selective screen erasing or graphics capability. The simulation model also needs 32K CPU minicomputer and BASIC software that is augmented with program chaining and sequential file addressing capabilities. Currently the package is implemented on an HP ACCESS system (Version 1B), with displays on superbee terminals. These are the minimum hardware and software requirements for the simulation model to retain its main features and advantages to the student.

## The CAI Package

The simulation model is coupled with seven tutorial lessons that teach the various spatial marketing and consumer behavior concepts, and then use the simulation model to illustrate the integration of the concepts into the whole. These tutorial units teach the concepts through the use of various story problems, simulation games, and inquiry dialogue, strenuously avoiding the use of drill and practice or simple presentation of material where at all possible. In this writer's experience, and



Would you like any other display? If so, which? [illegible]

Figure 3. The student has now added a node, added two links between the node and the markets, and deleted a link between two of the markets.

according to various studies,[2] such types of instruction are not particularly successful for teaching college undergraduates, nor do they make use of the unique pedagogical advantages of CAI. Further, the units are designed with enough flexibility to connect the concepts that the student is working with to his own experience as the concept is being taught. This was done both to illustrate to the student that these concepts work in the "real" world, and to help the student with retention of concepts by connecting the concepts with personal observations. Figures 5, 6, and 7 illustrate one portion of the first lesson which asks the student to apply his own experience to the concept of orders of goods. The student's own knowledge of where certain goods and services may be found is the foundation for his understanding of this concept.

Finally, the tutorial units allow the student to formulate his own definition of the concepts, instead of baldly stating the definition. This is done in a number of ways. For some simple definitions the student is merely asked to type in his own definition, or to complete a very incomplete definition using his own words. Then a key word search of the student's definition is made. If all of the key words are not found, the student is given the "correct" definition and asked to compare his against it, to see where he made his error. He is then asked if his definition contains all of the elements found in the "correct" definition. This is done because many wordings of the definition can contain implicit key words

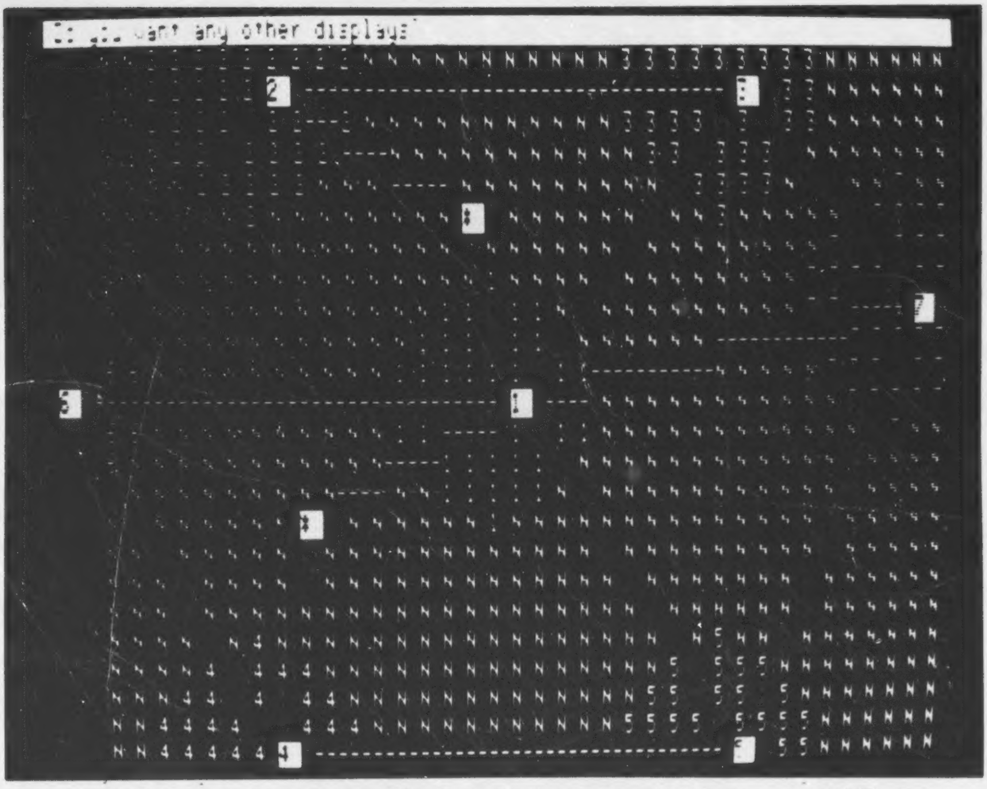
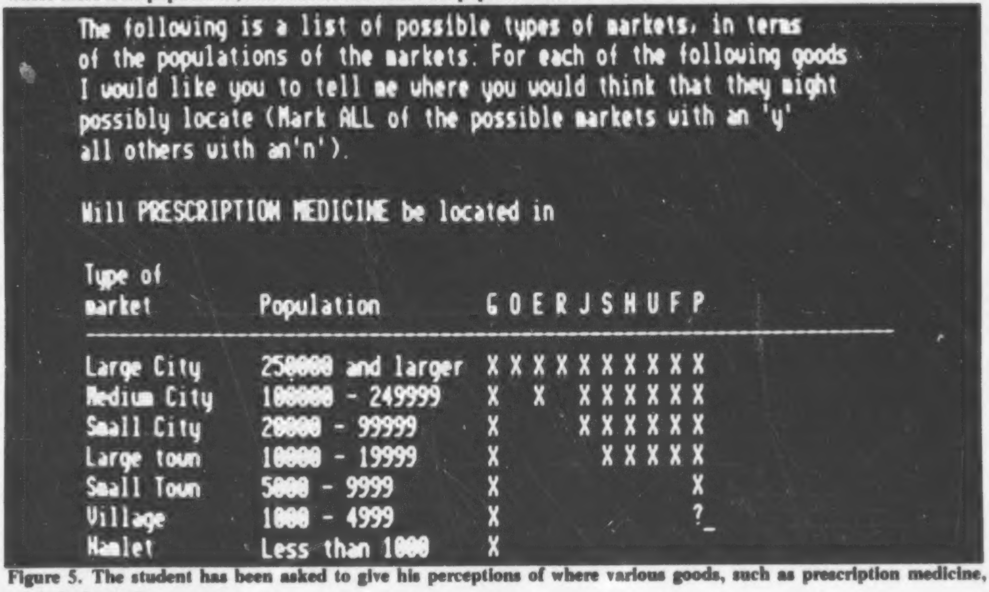
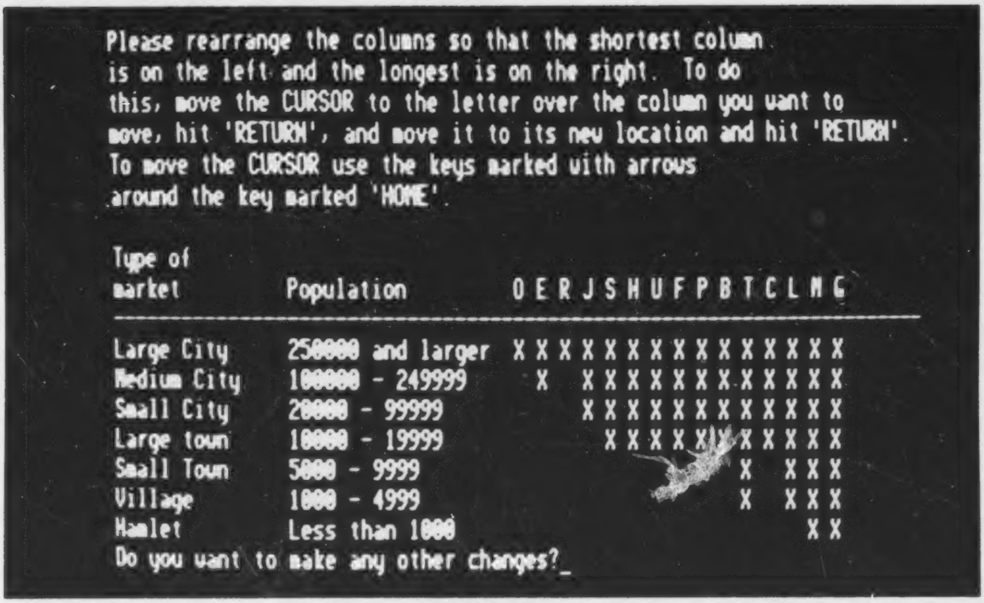


Figure 4. This display shows the market areas after the student has run the extended consumer model. The dots occur where there is no population, and N indicates where the population is not served.



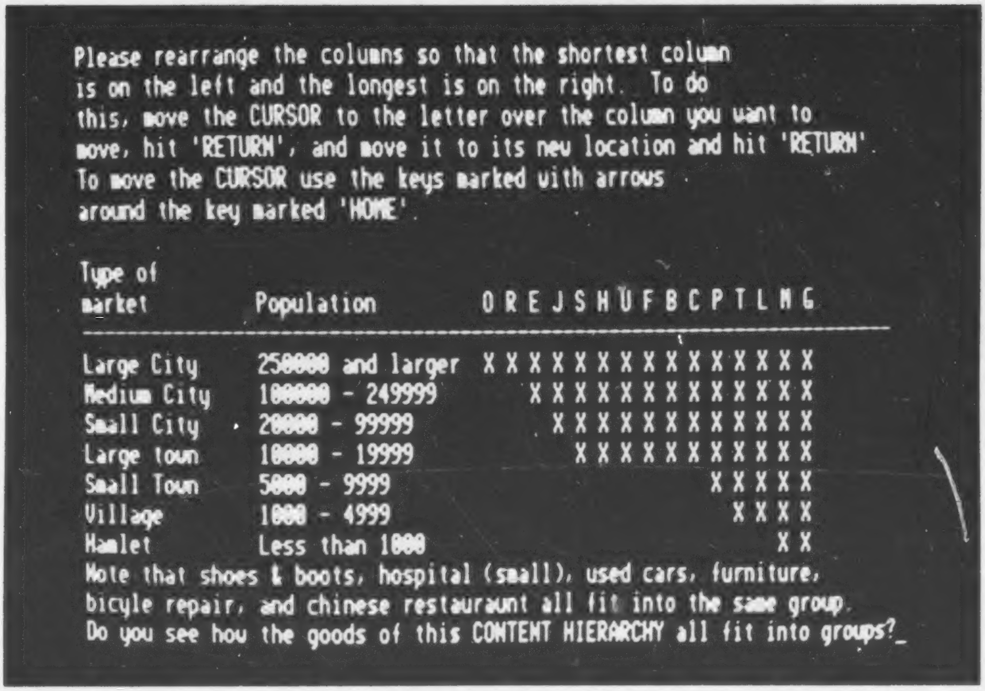
```
The following is a list of possible types of markets, in terms
of the populations of the markets. For each of the following goods
I would like you to tell me where you would think that they might
possibly locate (Mark ALL of the possible markets with an 'y'
all others with an'n').

Will PRESCRIPTION MEDICINE be located in

Type of
market         Population        G O E R J S H U F P
----------------------------------------------------------------
Large City     250000 and larger X X X X X X X X X X
Medium City    100000 - 249999   X   X   X X X X X X
Small City     20000 - 99999     X       X X X X X X
Large town     10000 - 19999     X         X X X X X
Small Town     5000 - 9999       X                 X
Village        1000 - 4999       X                 ?_
Hamlet         Less than 1000    X
```

Figure 5. The student has been asked to give his perceptions of where various goods, such as prescription medicine, might be located.



```
Please rearrange the columns so that the shortest column
is on the left and the longest is on the right.  To do
this, move the CURSOR to the letter over the column you want to
move, hit 'RETURN', and move it to its new location and hit 'RETURN'.
To move the CURSOR use the keys marked with arrows
around the key marked 'HOME'.

Type of
market          Population        O E R J S H U F P B T C L M G
-----------------------------------------------------------------------
Large City      250000 and larger X X X X X X X X X X X X X X X
Medium City     100000 - 249999     X   X X X X X X X X X X X X
Small City      20000 - 99999           X X X X X X X X X X X X
Large town      10000 - 19999             X X X X X X X X X X X
Small Town      5000 - 9999                               X   X X X
Village         1000 - 4999                               X   X X X
Hamlet          Less than 1000                                  X X
Do you want to make any other changes?_
```

Figure 6. The student has been asked to arrange the goods in order of their appearance in the hierarchy.



```
Please rearrange the columns so that the shortest column
is on the left and the longest is on the right.  To do
this, move the CURSOR to the letter over the column you want to
move, hit 'RETURN', and move it to its new location and hit 'RETURN'.
To move the CURSOR use the keys marked with arrows
around the key marked 'HOME'.

Type of
market          Population        O R E J S H U F B C P T L M G
-----------------------------------------------------------------------
Large City      250000 and larger X X X X X X X X X X X X X X X
Medium City     100000 - 249999       X X X X X X X X X X X X X
Small City      20000 - 99999           X X X X X X X X X X X X
Large town      10000 - 19999             X X X X X X X X X X X
Small Town      5000 - 9999                           X X X X X
Village         1000 - 4999                             X X X X
Hamlet          Less than 1000                              X X
Note that shoes & boots, hospital (small), used cars, furniture,
bicyle repair, and chinese restauraunt all fit into the same group.
Do you see how the goods of this CONTENT HIERARCHY all fit into groups?_
```

Figure 7. Finally, the student is asked to examine the graph to deduce the concept of content hierarchy.

that are difficult to search for, and because of the student's proclivity for unusual spelling errors. If the definition is rather more complex, i.e., containing several key elements, then a multiple choice strategy is used. Five definitions are given, of varying degrees of correctness. These include definitions that are wrong due to interchange of key elements, definitions that contain some but not all of the key elements, and of course the "best" definition, containing all key elements in the correct order. If the student chooses any incorrect definition, then the definition he chose is placed below the "best" definition, and an explanation is given as to why the definition chosen was not the best.

The concepts taught by these units fall into two groups: those that deal with the consumer's spatial behavior, and those that deal with the firm's decision on where to locate, which is based on both the consumer's behavior and on the non-uniform enviromental factors found in the simulation. The consumer's spatial behavior concepts include least effort behavior, behavior based on both the size of the market and the distance between the market and the consumer (the gravity model), multiple good shopping behavior, the effect of either convenience or shopping goods on spatial behavior, the action space variable, and the effects which culture and income have on the spatial buying pattern. The environmental factors include the location of other firms producing the same goods, all other firms, and variations in population density.

The CAI package has been constructed to be very simple for students to use effectively. The lessons are set up in page format (non-scrolling), so that all of the information that a student needs is displayed on the CRT. The student may stop at any point simply by typing STOP at any query. When the student restarts the program it will automatically start at the beginning of the page of material that the student was working on during the previous session.

## Applications

This package (the simulation and the CAI unit) may be used in a variety of geography classes and in other disciplines. Introductory human and economic geography, and introductory spatial analysis classes in geography, as well as classes in spatial economics and spatial marketing, would find the entire package valuable, and the material can be incorporated directly into the course. The simulation and/or CAI units can be used as enrichment material in other geography, economics, or marketing courses, e.g., cultural geography, to illustrate the cultural effects on consumers' buying behavior. Likewise, a course on consumer behavior could use the various consumer models as illustrations. The simulation by itself can be used by students in intermediate and upper level economic geography and marketing as tools to test their own ideas and concepts, and when used by a student who can program, the model can be used for various research projects and term papers.

## Notes

1. Paul R. Baumann, "Computerized Model for Teaching Spatial Organization," *Proceedings of the Conference on Computers in the Undergraduate Curricula 1974*, *Department of Computer Science, Washington State University*.

2. Judith Edwards, et al., "How Effective is CAI? A Review of the Research," *Educational Leadership*, November 1975, pp. 147-153.

# Simulated Study of Organizational Theory: The Use of EXPER-SIM in the Undergraduate Business Curriculum

**Paul Lorton, Jr.**
*College of Business Administration*
*University of San Francisco*
*San Francisco, California 94117*
*(415) 666-6771 or 666-6185*

## Introduction

Our knowledge of the world is tenuous at best. The more we study, the more we come to realize this. Many of us have spent considerable time learning and/or teaching probability and statistics to help us cope with the imperfect state of our understanding of the world around us. This is as true in the business world as it is in the world of physical, chemical or human behavior.

Business students, however, are especially determined to get the answer to their questions. They have little patience with the probabilistic ambiguity of the "real" world. Considerable effort is spent in giving them the statistical tools for such a world but very little time is available for them to actually assess the world with these tools. This is a special difficulty in courses dealing with organizational theory and problems.

It is not easy to provide a significant amount of data collection and analysis experience on ways in which different theoretical models can effect an organization within the confines of a one semester course. The purpose of the project described here was to provide this experience in a systematic fashion using the EXPER-SIM (Main, 1975) approach.

EXPER-SIM (Experiment Simulation) is an instructional approach (with supporting computer programs) which allows an instructor to build easily a theory based model of how data would occur if an experiment were actually conducted in a world where the theory held true. The data generated, when appropriately tested, will yield results typical of what one might find were the same study run in an unsimulated environment. The approach means data collection is a controlled, predictable process and the educationally important phases of hypothesis formulation and statistical interpretation can receive the student's and instructor's main attention.

Organizational Theory (BA 192a) and Organizational Problems (BA 192b) form a year long sequence of courses required of seniors in the College of Business Administration. Typically, three sections of about fifty students each take BA 192a in the fall semester and another single section of fifty students takes it in the spring. BA 192a meets for three hours of lecture per week with little integrated laboratory experience.

At this point, BA 192a and BA 192b tend to be discrete. The first course presents, in a variety of ways, theories of organizational behavior. In the second course (BA 192b), general problems of business are discussed often using cases and/or a management decision game played on a computer.

Using the EXPER-SIM approach, the emphasis in the Organizational Theory course could be on hypothesis testing. A set of experiences illustrating the process of experimentation and finding answers from a simulated world were developed. In general, the simulations concentrated on three contexts or problem areas: managerial style, worker incentive variants, and organizational structure.

With the EXPER-SIM capability, the students were able to experiment and actually develop testable hypotheses from which they could infer theories making the whole theory building process open to their experience.

The main reason for adapting EXPER-SIM to the Organizational Theory course was the same as the central purpose of EXPER-SIM itself. We wanted the experience in this course to give the students a fundamental familiarity with the design and evaluation of research. We feel that it is necessary to give the students a practical appreciation of organizational theories—that is, the student should be intimately aware of the process by which theories are developed. Too often, students read the theories of organizational behavior as fact—the use of EXPER-SIM in this course helped them appreciate theories as tentative summaries of tested hypotheses.

Ordinarily the Organizational Theory course is a lecture-text book course. Some attempts have been made to have the students experience hypothesis testing in this area first hand. Toward the end of the semester the students participate in a sequence of activities designed to let them study organizations first hand by being participant-observers. It is a popular element of the course but seldom produces testable data.

Trying to collect data about organizations from the "real world" has proven to be a difficult process which detracts from the purpose of the exercise because the focus of the effort quickly becomes data collection. Even with some data collected, there was often too little available for reliable illustration of the rest of the process. The use of the EXPER-SIM approach was intended to correct this situation.

### Design of the Project

During the spring, 1976, the LESS version (see Thurmond and Cromer, 1975a, 1975b) of EXPER-SIM was adapted to run on the Hewlett-Packard 2000E timesharing system in the College of Business Administration. A series of data generating contexts was developed for use in the Organizational Theory course. These contexts were designed to illustrate various theoretical positions discussed in the text for the course (Bobbit, et. al., 1974).

Five simulation contexts were used in all:

1. *Introductory Simulation.*

The purpose of this simulation was to introduce the whole process of simulating organizational behavior to the students. They could select various values for five variables. The relevant variables for this particular simulation were noise and lighting levels. Their effect was built to illustrate, in a very clean and emphatic fashion, the "Hawthorne" effect—whenever levels of these two variables were changed, productivity (one of the two possible dependent variables) went up.

2. *Simulation I: The Individual in the Organization.*

The second simulation was designed to illustrate the theoretical effect of various management initiated programs on job satisfaction for supervisors in a manufacturing environment. A variety of theories reported in the text discussion (see Bobbit, et. al., 1974, Chapter Four) of the individual in the organization were used. The results students achieved depended on what they selected to vary from the six possible factors, and whether they looked at scores of "satisfaction" or "dissatisfaction" as the dependent variable.

3. *Simulation II: Leadership*

The third simulation in the series was built on Fiedler's theory of leadership (Fiedler, 1967). This simulation was designed to reproduce the effects theorized by Fiedler in a fairly straight forward fashion.

```
GET-SEX.F
RUN

THIS SIMULATION WILL PRODUCE DATA FOR
EXPLORING THE WHOLE OPERATION ON A
FIRM AS PROGRAMS FOR VARIOUS DEPART-
MENTS ARE MODIFIED. (THE CURRENT STATE
IS NO TRAINING, NO COMPENSATION PLAN,
MIXED DECISIONS AND MIXED SCHEDULE. TO
GET A BASELINE RUN WITH THESE CONDI-
TIONS.)
```

```
DO YOU WISH INSTRUCTIONS (Y OR N) ?Y
INSTRUCTIONS:

  THIS SIMULATION PRODUCES THE EFFECT OF
VARIOUS   PROGRAMS   IMPLEMENTED   IN
VARIOUS DEPARTMENTS ON INDICES REFLECT-
ING THE OPERATION OF THE BUSINESS AS A
WHOLE. YOU CAN SELECT A VARIETY OF COM-
BINATIONS. IN EACH SIMULATION YOU WILL
SEE THE EFFECT OF YOUR MANIPULATION ON
PRODUCTIVITY, SALES, PROFIT AND NEW PRO-
DUCTS AND DEVELOPMENTS

THE FOLLOWING ARE THE VARIABLES AND
VALUES:


VARIABLE     VALUE
--------     ----------
DEPARTMENT   PRODUCTION / SALES /
             ACCOUNTING OR RESEARCH
TRAINING     PEOPLE / PRODUCTION / NONE
             (LOW,  MEDIUM  OR  HIGH
             LEVELS)
COMPENSA-    MORE, LESS, OR NO LINK TO
 TION        PROFIT, SALES OR
             PRODUCTION
DECISIONS    CENTRALIZED / DECENTRAL-
             IZED / MIXED
SCHEDULE     UNSTRUCTURED   /   STRUC-
             TURED / MIXED

INPUT VALUES FOR THIS RUN
DEPARTMENT                           ?P
TRAINING PROGRAM                     ?N
COMPENSATION                         ?N
DECISIONS                            ?M
SCHEDULE                             ?M
```

**Figure 1. Instructions and Sample Input for Final Simulation.**

4. *Simulation III: Managerial Grid*

This simulation was built in the context of the Managerial Grid (Blake & Mouton, 1964). It followed closely the form of the simulation above illustrating leadership style.

5. *Final Simulation: The Whole*

As part of an out-of-class final examination exercise, the students were assigned a very global simulation problem. The instructions for this assignment are appended. The actual simulation context was quite global. The main purpose of the assignment was to present a fairly complex situation that no one student could even begin to exhaust. A summary of the instructions for the context is also included in the Appendix.

The student signed onto the HP 2000E system and ran the appropriate program. Each simulation was designed to give the student all the information he needed to generate data. Interpretation depended on referring to the text and other students. Figure 1 is a sample of the instructions from the final simulation.

```
FINAL   SIMULATION   EXPERIMENTAL   RUN
NUMBER    1
CONDITIONS:
  DEPARTMENT       = P
  TRAINING         = N
  COMPENSATION     = N
  DECISIONS        = M
  SCHEDULE         = M


              ( IN DOLLARS )
          Units                        Units
Week      Prod      Sales    Profit    Develop
 1        119       168       61        43
 2        132       173       73        42
 3        118       131       50        39
 4        141       185       75        38
 5        162       176       81        32
 6        139       174       79        32
 7        142       149       71        35
 8        123       129       67        23
 9        125       170       77        27
10        135       182       65        26
11        135       137       76        33
12        161       125       56        41
```

**Figure 2. Sample Run #1 of Final Simulation**

After reading the instructions, the student selected the values for the variables he wanted to study. Figures 2, 3, and 4 show the results of varying training given in the production department. While engaged in one experiment, the program "remembered" what had happened before to allow for Hawthorne and other non-reversible experimental effects.

The students participating in this study were enrolled in one section of the Organizational Theory course during the fall semester, 1976. The students were assigned to run the simulation at approximately three-week intervals throughout the semester. They were encouraged to form small groups of two or three while running these simulations in order to facilitate informal discussion of which hypotheses to formulate and test.

```
FINAL   SIMULATION   EXPERIMENTAL   RUN
NUMBER 2
CONDITIONS:
  DEPARTMENT                           = P
  TRAINING                  = PE   LEVEL : H
  COMPENSATION                         = N
  DECISIONS                            = M
  SCHEDULE                             = M

              ( IN DOLLARS )
        Units                         Units
Week    Prod      Sales    Profit    Develop
  1      125       169       64        44
  2      139       174       72        42
  3      126       132       51        42
  4      153       184       78        38
  5      164       176       84        34
  6      146       176       82        31
  7      144       152       73        37
  8      131       130       68        26
  9      131       171       77        26
 10      147       181       68        25
 11      137       138       78        32
 12      162       127       58        42
```

**Figure 3. Sample Run #2 of Final Simulation**

```
FINAL   SIMULATION   EXPERIMENTAL   RUN
NUMBER 3
CONDITIONS:
  DEPARTMENT                           = P
  TRAINING                    = PR   LEVEL : H
  COMPENSATION                         = N
  DECISIONS                            = M
  SCHEDULE                             = M

              ( IN DOLLARS )
        Units                         Units
Week    Prod      Sales    Profit    Develop
  1      142       169       64        46
  2      152       175       73        43
  3      140       130       54        41
  4      157       183       80        39
  5      170       175       86        36
  6      164       176       80        30
  7      144       154       75        40
  8      140       130       70        29
  9      138       169       78        29
 10      154       182       70        25
 11      152       140       78        32
 12      177       130       58        45
```

**Figure 4. Sample Run #3 of Final Simulation**

For this report, the data collected consist mainly of responses to a questionnaire each student completed as he turned in his final work. Responses from about three-quarters of the class were analyzed for this report (the rest of the responses will be available too late to be included here).

### Results

During the course of the semester the students ran five simulations. A lot of class time was spent on the first three—discussing the appropriate statistical design and analysis, the range of possible findings, and actual findings of various members of the class. The fourth simulation was not discussed as fully but was used to reinforce the idea of building conclusions from the findings of a number of people. This was part of the required activity in the final simulation.

The students were then asked to rate each of these simulations at the end of the semester on the following scale:

0 Can't evaluate
1 Worthless
2 Marginal
3 OK
4 Good
5 Excellent

Table 1 lists the percentages responding in each category and the mean response to each item.

The students were also asked to rate three other aspects of the course: the text, the final exam activity, and the group activity described above. Table 2 gives a summary of responses to these items.

Finally a series of yes/no questions was asked to assess the role the simulation played in the course in general. The questions were:

1. Do you feel the simulations contributed to your appreciation of earlier courses?
2. Do you feel the simulations were valuable?
3. Do you feel the simulations were used appropriately?
4. Do you feel the simulations helped you evaluate the group project?

**Table 1. Percentage Responding in Each Category and Mean Response in Rating Five Simulation Activities.**

| | *Rating* 1 | 2 | 3 | 4 | 5 | Mean |
|---|---|---|---|---|---|---|
| 1. Sample Simulation | 4.7 | 9.5 | 23.8 | 47.6 | 14.2 | 3.57 |
| 2. Simulation I | 4.7 | 9.5 | 33.3 | 38.0 | 14.2 | 3.47 |
| 3. Simulation II | 5.0 | 10.0 | 30.0 | 35.0 | 20.0 | 3.55 |
| 4. Simulation III | 13.6 | 9.0 | 36.3 | 31.8 | 9.0 | 3.13 |
| 5. Final Simulation | 5.0 | 5.0 | 25.0 | 25.0 | 20.0 | 3.50 |

**Table 2. Percentage Responding in Each Category and Mean Response in Rating Non-Simulation Components of the Course.**

| | *Rating* 1 | 2 | 3 | 4 | 5 | Mean |
|---|---|---|---|---|---|---|
| Textbook | 9.0 | 9.0 | 27.2 | 50.0 | 4.5 | 3.32 |
| Final Exam | 0.0 | 4.7 | 38.0 | 42.8 | 14.2 | 3.67 |
| Group Activity | 0.0 | 4.7 | 33.3 | 23.8 | 38.0 | 3.95 |

Table 3 gives the percentage responding yes/no to each of these questions.

**Table 3. Percentage Responding Yes/No to Four Questions**

| | *Yes* | *No* |
|---|---|---|
| Question 1. | 71.5 | 28.5 |
| Question 2. | 80.9 | 19.1 |
| Question 3. | 80.9 | 19.1 |
| Question 4. | 66.7 | 33.3 |

## Discussion

The results of this effort are quite rewarding. One can see two things very clearly: (1) the students are using their statistics training to test hypothesis and, after considerable review, are actually doing a credible job, and (2) the students are gaining a clearer idea of what a theory is and how it comes about. In addition, from the responses to the questions presented above, the students were quite positive about the experience.

To see students actively and intensely engaged in collecting data, analyzing the data, and exchanging conclusions is most rewarding. Some students actually realize this, see the purpose of the exercise, and are quite involved. Other students don't see the point and go through the exercise as an exercise.

This latter type is troublesome. One can see that they are profiting and learning from the experience, but they are still looking for definitive answers and not for methods for finding answers.

Part of the problem and part of the eventual solution for these students, is to introduce them to the tentative nature of the world and statistic's role in that world in their freshman statistics course. Perhaps they spend too much time doing the problems in the back of the chapters in the statistics book and too little time getting data from the real world where things don't always come out even (or worse yet—a perfect square).

One interesting incident in the course of the project was the necessity for devoting the better part of four class periods to a review of statistics. All the students in the course had completed the four semester quantitative methods sequence required by the College. This sequence included two semesters of statistics and a substantial amount of probability in a third course.

The students seemed to have a tolerable grasp on the mechanics of various statistical techniques—that is, they could take two groups of independent numbers and produce a value for a Student's t. Since programs available for easy use on the computer did the mechanics of the analysis, this computational skill was not needed or used. The needed skill was in design and interpretation. Forming a testable hypothesis and testing it with the computer doing the analysis were the needed skills.

The experience was well worth it and the results are compellingly positive. The use of EXPER-SIM as a system for devising data-generating contexts is quite useful in many business college settings. The technique will be used in other courses.

Certainly the later stages of the introductory statistics course can benefit from the use of EXPER-SIM. Marketing research, managerial psychology, and even operations management are all courses for which EXPER-SIM applications will be developed.

## Appendix

### Final Simulation: Instructions

You will be given the specifications for the final simulation. For this final simulation you are to do the following tasks:

1. Test a particular hypothesis which seems interesting to you.
2. Write a brief summary of your hypothesis and findings.
3. Exchange the above with other investigators. Be sure to demand quality for quality—that is don't give a nicely typed document in exchange for something written on the bottom of a Dixie Cup.

This should result naturally in the following report of three to five pages:

1. Introduction to the aspect of the situation you explored (half a page or so).
2. Statement of your hypothesis indicating the theory or theories to which it is related (half a page or so).
3. Statement and interpretation of your results (about one page).
4. Statement of the results of other investigators and how they relate to yours. (Bonus points awarded for each "other investigator" cited.) (Variable length depending on how many you get.)
5. Brief summary, conclusion and suggestions for further work developing out of sections one through four above. (A paragraph or so of each summary, conclusion, suggestion.)

You may do this in groups of no more than THREE (3). If the group size is more than ONE (1), the report MUST be typed and turned in clearly indicating the members of the group who participated. It can be turned in apart from the final but you are best off by turning in all finals and the report at the same time.

### Final Simulation

You are in charge of an organization. You have complete control of what the organization does. While this sounds ideal and unrealistic, it is not since your real control is limited to doing some very global things and evaluating their effect on some rather broad indices. Essentially you have several departments and a series of indices you are watching.

The departments are:
- Production
- Sales and Marketing
- Accounting and Finance
- Research and Development

The indices you are watching include:
- Productivity
- Sales
- Profit
- New Products and Developments

For each run of the simulation you can implement a program in a department and watch its effect over 12 weeks on each of the indices. The programs you can implement are:

*Training Supervisors.* You can implement programs which will make supervisors in any of the departments low, medium or high on concern for production or concern for people.

*Modification of Compensation Plan.* You can make compensation linked more or less directly to profit, sales or production.

*Modification of Decision Making Locus.* You can make decision making highly centralized, highly decentralized or mixed within any of the departments.

*Modification of Work Schedule.* Working hours can be made unstructured, highly structured or systematically flexible (a function of quotas, etc.) in any of the departments.

The final question for you is how to improve the organization's operation. What improvement consists of is up to you to define. What specific area you choose is also up to you. Focus on one aspect and explore an effect of one program. Depend on others to attack other areas (coordinating your efforts with them for best results).

**Acknowledgment**

The study reported here was made possible by a grant from the Exxon Educational Foundation.

## References

Blake, R. R. and Mouton, J. S. *The Managerial Grid.* Houston: Gulf Publishing Co., 1964.

Bobbit, H. R.; Breinholt, R. H.; Doktor, R. H and McNaul, J. P. *Organizational Behavior: Understanding and Prediction.* Englewood Cliffs: Prentice Hall, 1974.

Fiedler, F. E. *A Theory of Leadership Effectiveness.* New York: McGraw Hill, 1967.

Main, D. B. "An Overview of the EXPER-SIM (Experiment Simulation)" Pedagogy. In Lecarme, O. and Lewis, R. *Computers in Education.* Amsterdam: North-Holland Publishing Co., 1975.

Thurmond, J. B. and Cromer, A. O "Models and Modeling with the Louisville Experiment Simulation System (LESS)." *Behavior Research Methods and Instrumentation,* 1975, 7, 229-232 (a).

Thurmond, J. B. and Cromer, A. O. *LESS: Manual for Users of the Louisville Experiment Simulation System.* Univ. of Louisville, 1975 (b).

# Using Macroeconomic Simulations in the Principles Course for the First Time

**David S. Moewes***
*Department of Economics*
*Concordia College*
*Moorhead, Minnesota 56560*

**Currently on leave to:*
*Department of Economics*
*University of Utah*
*Salt Lake City, Utah 84108*
*(801) 581-5631 or 487-5570*

### Introduction

Teachers with little or no previous computer experience who would like to use macroeconomic simulations in their principles classes face numerous difficulties which may be more than they really bargained for. If they decide they would rather not get into programming the simulations, then somehow they must find out how and where to obtain suitable existing programs. If computer resources are limited, as they often are at smaller colleges, then there are additional problems in modifying and adapting such programs to their own computer system. If, on the other hand, they would prefer to use a simulation which corresponds as closely as possible to the texts and techniques currently being used in their classes, then they must learn to program or find someone else who is willing to do it for them. Either way, the use of a simulation is probably going to require a substantial commitment on the part of the teacher.

The experience gained over the last four years through the use of the macroeconomic simulation which is described in this paper suggests that maximum learning effectiveness will more likely be achieved if the simulation is programmed by the instructor. Such an approach will give the teacher the flexibility that is needed to meet the inevitable problems that somehow seem to develop. Through continual modification, based on student performance and comments, the ultimate potential of a simulation can be attained. Unfortunately, the large initial commitments of the programming approach may bias first-time users against using it.

For many economics students their first exposure to computer simulations comes in the form of sophisticated analytical or predictive models. The complexity of these models tends to limit their learning effectiveness for many beginning principles students who are unable to appreciate the model's theoretical subtleties. Other students have had to use very abstract and simplistic models which are utterly devoid of any empirical richness whatsoever. Certainly both types of models have their purpose and each is capable of helping students understand macroeconomic theory, but an optimal simulation for principles classes appears to be one which strikes a compromise between too much and too little empirical detail.

An optimal computer simulation for this level course should be one which is relatively easy for students to understand and use; one which complements what the students are learning in their lectures; one which uses the same policy variables and terminology as their textbooks; and hopefully, one which stresses the theoretical concepts that the teacher feels are particularly important. It should also give the student an appreciation for the complexities and interrelationships that exist in an economic system without being overly complex itself. Perhaps even more important, the simulation should be one with which the teacher feels comfortable. Students who are using the computer for the first time tend to be full of fears—some real and some greatly exaggerated—about computers, what they're really all about, and how to use them. Having encountered several situations where teachers have relied on "outside" computer help to explain and implement their simulations, we have found that it is extremely important for a teacher to be able to answer the various questions about accessing the model on the computer and the operation of the model itself as those questions arise. Postponing answers until others are able to respond to them invariably causes problems for the students and leads to a lower level of effectiveness for the simulation.

## Description of the Model

When time-sharing computer facilities were installed on our campus the author decided to try and develop a macroeconomic simulation, which after many modifications has come to be called MACRO4, that would allow students to apply, in a fairly realistic setting, the theoretical concepts that they had learned through class lectures or through their readings. The MACRO4 model is basically an adaptation of the standard IS-LM model which incorporates an aggregate Cobb-Douglas production function and a Phillips curve.[1] It has just three policy variables—government spending, the rate of taxation, and the supply of money—which the policy-maker (user) must adjust as he or she attempts to provide full employment, stable prices, and maximum economic growth (as measured by real GNP per capita) for a hypothetical economy which simulates in many ways the operation of the U.S. economy.

The simulation was designed to emphasize the impact and interrelationships of the policy variables, the growth requirements that are implicit in an economy with a steadily increasing labor force, the differences between real and nominal variables, the difficulties of attempting to fine-tune even a relatively simple economy, and the trade-offs that must be made between conflicting economic goals. (A second, more complex, version of MACRO4 was later developed which included a foreign sector, and it was designed to illustrate the problems involved in maintaining balance of payments equilibrium and the trade-offs which must be made between domestic economic goals and foreign sector goals.)

Through manipulation of the three policy variables, students can control total aggregate demand in the economy. Their main objective is to bring aggregate demand into rough equality with the productive capacity of the economy. The productive capacity is largely independent of policy decisions, at least in the short-run, and it is determined by the amount of capital stock (which the user can marginally affect through changes in gross investment spending), the size of the labor force, and the level of technology. If total spending is greater than productive capacity, then the economy will experience a higher rate of inflation and unemployment will be relatively low. The opposite situation, where there is too little spending relative to productive capacity, brings about recession conditions which are characterized by higher levels of unemployment and lower rates of inflation. In terms of the model's structural relationship between inflation and unemployment, the user has moved downward and to the right along the Phillips curve. The structural specifications of the model are shown in Table 1.

Included in each student's set of instructions is a simulated history which contains the results of six previous decision-making rounds. It records the performance of a notoriously inept previous administration which has gotten the economy into rather bad shape. The students' administration begins on the seventh round and they must immediately decide how to overcome the

**Table 1. Specification of the Model**

| | | |
|---|---|---|
| $C = 27.6 + 0.92\,Y_d$ | | Consumption Function |
| $Y_d = 0.85\,Y - T$ | | Disposable Income |
| $T = -94.5 + t_x\,Y$ | | Tax Function |
| $I = 10 + 0.2\,Y - 1200\,i$ | | Investment Function |
| $Y = C + I + G_o$ | | Commodity Market Equilibrium |
| $M_d = 0.25\,Y + 1/i$ | | Demand for Money Function |
| $M_d = M_s$ | | Money Market Equilibrium |
| $L = 1.015\,L_{-1}$ | | Labor Supply Function |
| $K = 0.97\,K_{-1} + I$ | | Capital Stock Function |
| $A = 1.02\,A_{-1}$ | | Rate of Technological Change |
| $Q = A\,L^{.7}\,K^{.3}$ | | Production Function |
| $U = 0.025 + \dfrac{P_{-1}\,Q - Y}{2.5\,P_{-1}\,Q}$ | | Rate of Unemployment |
| $R = 0.08 - U$ | $[\text{if } U \geq 0.06]$ | Rate of Inflation |
| $R = 1.1\,(0.08 - U)$ | $[\text{if } U \geq 0.03]$ | |
| $R = 1.5\,(0.08 - U)$ | $[\text{if } U \geq 0.015]$ | |
| $R = 1.75\,(0.08 - U)$ | $[\text{if } U \leq 0.015]$ | |
| $P = P_{-1} + R$ | | Price Level |

Notation:

$C$ = Consumption Spending
$T$ = Tax Receipts
$I$ = Investment Spending
$G_o$ = Government Spending
$M_s$ = Supply of Money
$Q$ = Output Capacity
$U$ = Rate of Unemployment
$R$ = Rate of Inflation
$P$ = Price Level
$P_{-1}$ = Last Round's Price Level

$Y$ = National Income
$Y_d$ = Disposable Income
$t_x$ = Rate of Taxation
$i$ = Interest Rate
$M_d$ = Demand for Money
$L$ = Labor Supply
$L_{-1}$ = Last Round's Labor Supply
$K$ = Capital Stock
$K_{-1}$ = Last Round's Capital Stock
$A$ = Level of Technology
$A_{-1}$ = Last Round's Level of Technology

economic problems caused by the previous administration. At the beginning of each round, the user must answer three questions concerning the previous round's value of capital stock, price level, and investment spending so that the computer can determine the user's starting point. This is necessary because the model is programmed in such a way that if a student has to quit before the computer assignment is completed, then the student can come back later and start where he or she left off. The user then inputs the values for three policy variables and the computer prints out that round's economic results, a sample of which is shown in Table 2.

An hour of class time is spent explaining the structure and operation of the model. Depending upon circumstances, another hour is sometimes spent discussing the computer and how to access it. However, even after going through the five pages of instructions that accompany the simulation most students feel quite bewildered by the various aspects of the model, even though the economic concepts are quite simple. The vast majority of students do quite well after they have played a couple of experimental rounds. One minor problem that has developed with this particular simulation is that students expect too much of themselves initially and some tend to get discouraged if they don't obtain acceptable results with their first try at using the model. They are cautioned that they should experiment with various types of fiscal and monetary policy so that they can get the "feel" of the model before they start their assignment. The average time spent on the time-sharing consoles is about two and one-half hours, but that varies considerably depending upon the student. Another hour or two must then be spent answering a set of policy decisions which are based on their experience with the model.

**Table 2. Sample Output**

```
What period is it?   7

What was last period's value of capital stock?  2,047
What was investment spending last period?   150
What was the price level last period?   1.12

What is this period's level of government spending?   230
What is this period's rate of taxation?   0.3
What is this period's supply of money?   280

Given the decisions which you made this round concerning G, T, and M the economy achieved the following results:

Current population = 164.02 million people
Total labor force = 82.01 million people

Current dollar GNP (the level of spending) = $1,028.2 billion
Current dollar GNP per capital = $6,268.61
Real GNP = $892.71 million
Real GNP per capita = $5,442.80
Current rate of unemployment = 5.12 percent
Current price level = 1.15
Rate of inflation = 3.17 percent
Rate of interest = 4.36 percent
Current consumption spending =$634.79 billion
Current investment spending =$163.37 billion
Net government income tax collected = $213.95 billion
Government spending was $230 billion
You are running a budget deficit of $16.05 billion
The amount of capital stock changed to $2,135.59 billion
Your administration's total budget deficit or surplus after one round = $ -16.05 billion

What period is it?
```

Note: Student responses are underlined.

Conversational remarks (mostly humorous) have been inserted into the program to make the simulation more interesting for the user. The MACRO4 simulation has a swearing-in ceremony in which the user is asked if he or she pledges to uphold the laws of supply and demand while in office. Other remarks are printed out when a particular round's results do not meet certain prescribed standards. For example, housewives start to get upset when the rate of inflation gets above six percent and the army of the unemployed starts to march when unemployment becomes excessive. The computer's remarks have contributed to the success of the simulation and have certainly helped to change the impersonal image that is often associated with computers and other educational technology.

## Problems

One of the ongoing problems associated with MACRO4 has been whether to allow students to replay each round until they get acceptable results. The model is currently programmed so that individual rounds can be repeated and that encourages policy experimentation by users. The alternative of not allowing them to do this has the advantage of making them live with their mistakes and it emphasizes the non-reversible nature of most policy decisions. It does, however, increase the amount of computer time necessary to complete the project since students have to go all the way back to the initial round if a serious mistake or typing error is made.

Another problem that has arisen when the simulation has been used in larger classes is the copying of other students' results. Although working together on the simulation has not been expressly prohibited in the past because it was felt that learning could take place under those conditions, there are always a few students who insist on abusing that type of arrangement. With respect to MACRO4, two different methods are available for making the simulation perform differently for each user. The first, and possibly the easiest method, would be to add stochastic error terms to several of the structural and behavioral equations in the model. The consumption and investment functions, the Phillips curve, and the labor force and technological change functions could be easily modified in this way. A possible drawback to this approach is that it would make some of the model's basic economic relationships less precise and therefore less predictable. In one sense such changes would make the model more realistic since the real world is not as predictable as implied by the structural equations of the model, but these changes could conceivably reduce the simulation's effectiveness in illustrating theoretical concepts. A second method would be to randomize each user's starting point.

The most serious problem encountered so far has been the insertion of the simulation into the principles course structure. Most principles courses, including the author's, are structured in such a way that it makes it difficult to utilize the simulation in an optimal way. It should be used right after discussing the Keynesian macroeconomic model and concurrently with discussions of fiscal and monetary policy. Unfortunately, both these discussions usually occur at the midpoint of a one semester class, or even later in some instances, and introduction of the simulation at that time would not provide enough time for all students to have reasonable access to the computer. Thus, the simulation is usually used before the students have an adequate theoretical background for understanding the model, and the model must be discussed in rather intuitive terms. This seems to have had more of an impact on students who are already having difficulties with the course, but it has not been a significant factor since they can compensate by spending more time on the simulation. Restructuring the course has been considered but no serious movements have been made in that direction.

## Conclusions

Although no formal testing has been undertaken to determine the simulation's effectiveness in helping students learn macroeconomic theory, all available evidence suggests that it is quite helpful. It would seem that making available to students an alternative learning mode, for a part of the course which has traditionally been considered difficult, would help their conceptual understanding of that material.

Student comment and reaction has generally been favorable and in some instances very positive and enthusiastic. Some of their remarks are shown below:

> I got a better understanding of how C, I, and G effect [*sic*] our economic situation and also a better understanding of the relation between inflation and unemployment. It would have been better if the instructor had started on the macro section of the text to give us a better background to the problem at hand.
>
> I learned more about the United States and the President's [*sic*] job than I would have if I would have listened to 10 lectures on this subject. This so called "playing with the computer" was a very valuable experience and one that I will be able to remember along with the facts from it.
>
> I was able to put book learning into practice. This is something I will always be able to relate to and not just a bunch of words. I was able to understand the concepts much more because of this. My suggestion—more of this, putting into practice what we learn in books.
>
> I learned to be a little more patient with politicians' efforts to keep the economy on an even keel.

In the final analysis, the teacher must decide whether the time spent in developing a simulation and the time spent in class discussing it are worth the educational benefits that the student receives. Are there other alternatives or learning modes that could be used instead of the simulation which could accomplish as much—given the same amount of teacher and student time expended? The author's experience with MACRO4 and other computer projects indicates that such projects constitute the most effective part of this principles course.

## Acknowledgments

Part of my work on the MACRO4 simulation was funded by a 1972 summer grant from the College Science Improvement Program (COSIP) and by a 1973 Bush Foundation faculty fellowship. The financial support provided by them is gratefully acknowledged.

I would like to thank Mary Patterson and Wayne Stevenson, Department of Economics, University of Utah, and Arne Garness, Director of Concordia College's Computer Center, for their generous help, patience, and encouragement.

## References

1. Several aspects of the model were initially suggested by a two-sector model which was described in Richard E. Attiyeh's article, "A Macroeconomic Model for the Classroom," which is found in *New Developments in the Teaching of Economics*, Keith G. Lumsden, ed. (Englewood Cliffs: Prentice-Hall, Inc., 1967).

# Computerized Gaming and Simulation in Higher Education: The METRO-APEX Experience

**Jane Wolin**
*Center for Computer and Information Services*
*Rutgers, the State University*
*New Brunswick, New Jersey 08903*

## Abstract

Simulation is an educational technique which allows students to learn by participating rather than by listening or watching. There are many different types of simulation games available to educators some of which use the computer to [illegible]lyze pla[illegible] decisions and calculate results.

METRO-APEX is a computer-based gaming simulation in which students study a large urban city and its surroundings by playing the roles of city and county politicians, industrialists, environmental control officials, planners, land developers, and labor groups. Each cycle represents one year in the city. At Rutgers METRO-APEX was played once a week for five weeks during the fall of 1975 as part of a one credit course. The decisions made by the players were coded at the end of each session and used as input to the METRO-APEX computer program. The output from the program told each player what had happened to the city during the year as a result of these decisions and served as a basis for the next session.

This paper will discuss METRO-APEX as a specific example of how simulation and the computer can be used to further the learning experience in the undergraduate curricula.

During the fall of 1975, Professor David Rosen of the Department of Human Ecology and Social Science at Rutgers University offered a one credit course entitled "Lab. in Political Science." (The course is being offered again during the spring of 1977.) The basis of this course is a computerized simulation called METRO-APEX which is described below. Because Professor Rosen had very little computer background or experience, a cooperative arrangement was made with the University's Center for Computer and Information Services (CCIS), and this writer was assigned to the project. All of the computer aspects of the simulation were handled by CCIS including putting up a new version of the METRO-APEX program on the system, creating all the necessary files, overseeing the preparation of the input data, doing the actual processing, detecting errors in the output and making the necessary corrections, etc. Professor Rosen served as the Game Overall Director (G.O.D.), was in charge of the "gamed" environment as described below, conducted the all-important debriefing session, assigned and graded student papers and in effect dealt with all the aspects of learning and education.

## The METRO-APEX Simulation

METRO-APEX is a "computer-based gaming simulation. . . designed to provide a laboratory urban community in which air pollution management trainees. . . apply and test the knowledge and skills gained through conventional educational methods." [1] It evolved from a similar exercise (M.E.T.R.O.) developed at the University of Michigan. In 1966, the Division of Air Pollution Control, U.S. Public Health Service, awarded a grant to the COMEX Research Project, University of Southern California, to expand the original work. The universities worked together to produce the initial METRO-APEX simulation in 1971, as well as a revised and updated version released in 1974. [1]

METRO-APEX is a simulation rather than a game. It analyzes and interrelates players' decisions according to a very complex model with the use of a computer. Players are not directly pitted against one another in order to determine a winner. (The minor exception to this is the election model which may or may not be used at the discretion of G.O.D. See below.)

In effect there are two essential components to METRO-APEX. One is the "gamed" environment during which the players assigned to the various roles interact with one another and make decisions. Each of these sessions (or cycles) lasts approximately three hours and represents one year in the life of Apex county.

The second component is the group of computer programs which "predicts the changes that occur in several sectors of the urban system in response to the decisions made by participants in the 'gamed' environment, decisions made by persons outside the 'gamed' environment (other actors whose behavior is simulated in the computer), and external pressures on the metropolitan area (also simulated in the computer)." [1] The computer analyzes the decisions made by the players, calculates the changes in the status of the community and generates output. This output in the form of tables, maps, summary reports, financial statements and a newspaper is given to the players who use the information as the basis for the next cycle.

The roles to which the players are assigned are as follows:

1. City and county politicians whose job it is to administer their respective jurisdictions. They prepare budgets, allocate funds, pass laws and initiate and manage programs.

2. City and county planners who assist politicans and develop long-range plans by which the community may be guided.

3. Environmental Quality Agency which consists of an air pollution control officer, a solid waste manager and a water quality manager. These people formulate policy, monitor the levels of pollution and are responsible for "cleaning up" the county.

4. Land developers who buy and sell land, and build homes, apartment buildings and commercial establishments and attempt to sell them at a profit.

5. Industrialists who manage their individual companies. They set prices for their products, determine production levels, select fuels, decide on pollution controls, negotiate with union leaders, declare dividends to stock holders and are generally responsible for seeing to it that their companies make money.

6. Pressure groups such as the Tax Payers Association and the Minority Action League who urge the politicians to pass legislation according to their needs and who attempt to gain support for their projects from the other players.

7. News Media which advocates or disapproves of government decisions, sets editorial policy, reports the activities of the players and covers all the important news stories.

8. Labor Council which annually negotiates labor contracts with each of the industrialists. The Council proposes annual salaries, number of sick days and vacation days, health insurance coverage, retirement fund contributions and other fringe benefits. If labor and industry cannot reach an agreement, they may consent to binding arbitration, the union may strike or management may conduct a lockout. (Note: The labor model is not part of the METRO-APEX package from COMEX. It was designed and written by Alex Jones of Princeton University and is a fine addition to the simulation.)

For each of the roles outlined above there must be a role advisor who serves in several capacities. This person meets with his or her players at the beginning of the first session to give detailed instructions on the rules under which the players must operate. The advisor describes and outlines the roles of the players and explains how to fill out the work sheets. (The work sheets, which are unique for each role, are designed to help the players put their decisions on paper.) During each cycle, the role advisor answers questions and generally oversees the functioning of the players. At the end of the cycle the advisor codes the information from the work sheets to special coding forms. The data from these forms are later keypunched and used as input to the computer programs.

Overseeing the entire process of the simulation is the Game Overall Director (G.O.D.). This person begins the first cycle with a short talk outlining the general rules, introduces the various roles and describes the geographic layout of Apex County. During each cycle the game director must constantly make rulings and settle disputes. This usually requires simulating additional roles such as labor arbitrator, judge, sheriff, state legislature, governor, Federal Government, etc. The director may assume some of these roles or assign them to the role advisors.

It is impossible to detail all aspects of this simulation in these few pages (the METRO-APEX manuals total 21 volumes). By describing its use at Rutgers University, perhaps the reader will be able to visualize how it works.

Thirty-five undergraduates registered for the course which was scheduled to meet one evening a week for six weeks. Each student was assigned a role and given some introductory material a few days before the course began.

When the students arrived, they were given tags which contained their names and roles. Several large tables were set around the room with signs indicating the location of each of the various groups (e.g., developers, county politicians, etc.). Copies of the *Metro-Apex News*, the local computer-produced newspaper, were distributed as well as a computer printout applicable to each role which had been created from a preset initial cycle. After the preliminary introductions from G.O.D. and the role advisors, a prerecorded news broadcast was played on the tape recorder, and the simulation officially began.

For the first hour or so things were fairly quiet as the students familiarized themselves with their roles. They read the newspaper which described several issues requiring a vote, and they completed their opinion polls. Soon labor and industry began negotiations, the Environmental Quality Agency pushed for legislation requiring industry to meet certain minimum standards with sanctions if they refused, the developers began negotiations with the county politicians to build low-income housing and the planners proposed zoning changes.

After the session ended, the coding was completed, news articles were collected and the tabulations from the opinion polls were recorded. The following day after all the data had been keypunched, the computer work began. The initial program run indicated errors on some of the input cards, and careful scrutiny of the output revealed inconsistent results (negative balances, unpaid taxes, etc.). The input cards were changed and many of the errors were corrected. However, it took two more runs before acceptable output was obtained.

When the students returned the following week, they were anxious to see the computer output and to find out what had happened. During the second cycle G.O.D. announced that elections would be held during cycle four, and the results would be announced in cycle five. All incumbent politicians chose to run again and a slate of opposing candidates was drawn from among the other players. Campaigning began and campaign contributions were solicited all during the next two cycles. This, of course, went on concurrently with the rest of the simulation.

Because of the complexity of the model, no two METRO-APEX simulations are alike, and many interesting issues and projects can and do arise. During the Rutgers simulation, for example, the county hospital facilities were expanded, a solid waste treatment plant was constructed, hearings and litigation resulted when Shear Power Company requested a rate increase, and a master plan was devised and accepted which included land usage, capital improvements, environmental control, etc. Also, during one cycle a county politician sued the *Metro Apex News* (played by this writer) for libel and slander. Lawyers were hired, a judge was appointed to hear the case, and all had their day in court.

A final example of learning from experience occurred with one of the industrialists. His brewing company had done so well and had made so much money, that during cycle two he purchased land and expanded his plant facilities. In cycle three he increased his production and came back the following week expecting that he had made a fortune. However, he overestimated the market capacity and instead of selling nine million barrels of beer, he sold five million. Naturally his raw materials costs were up, as were labor and fuel costs. In addition, he now had to pay more than $1 million to carry his inventory over from one cycle to the next. Instead of having a $5 million cash reserve, he discovered that his company was $4 million in debt—but he sure had lots of beer on hand!

During the course of the simulation, the players are led to believe that they will participate in more cycles than actually will be played. This is to prevent wild, speculative decisions from being made during the last round. When the players arrive, for what they believe is the last cycle, they discover that the simulation has ended, and a debriefing is conducted.

Perhaps the most important aspect of a simulation is the debriefing. At this time the players and G.O.D. discuss and review what has taken place. The theories and concepts underlying the model are brought out, and the model itself is critiqued as to how well it relates to the real world.

When played at Rutgers, the course was scheduled to last six weeks, but at the beginning of cycle five the prerecorded news broadcast prepared by G.O.D. announced that Apex County had been completely destroyed by a nuclear blast! During the ensuing debriefing session, three very interesting results emerged. After each of the players briefly recounted his or her experiences, the group was amazed to learn that there were vast areas of ignorance and misconceptions concerning the roles of the other players. The industrialists and the Labor Council were totally unaware of what the county politicians were doing. The land developers had no idea of problems confronting the Environmental Quality Agency, etc. This was then related to real-life situations where individuals are often so involved with the "trees" that they are unable to see the "forest."

Secondly, the students discovered that within the group there were grave perceptional differences of the same event. There were wide discrepancies among the various reports of "what happened."

Lastly, during a critical analysis of various aspects of the model, the students realized that often what appeared at first glance to be a flaw was actually a true representation of the real world. A striking example of this was the air pollution control officer who claimed that simulated aspects of the model caused apathy among the politicians in passing strong legislation and resulted in the resistance of the industrialists in installing pollution controls. The other players, however, quickly pointed out that these very factors have prevented most major industrial areas in the country from solving their pollution problems.

Although the students were overwhelming in their praise of the simulation as a teaching technique (see evaluations below), from an administrative point of view there are difficulties in using METRO-APEX. One factor essential to success is a knowledgeable and qualified staff. This includes:

1. Experienced role advisors (preferably people who have participated in METRO-APEX previously).
2. An experienced keypuncher (ideally one who is familiar with the control cards required by the program).
3. A computer person (or two) who is completely familiar with data management, JCL, the program and the output, and who understands the interrelationships of the roles (e.g. a miscoded value on the county politicians' budget card will cause the decisions of the air pollution control officer to go awry).
4. Additional personnel to duplicate, staple, separate computer output, post maps and tables, help set up and clean up, and generally see to dozens of details.
5. A game overall director who understands the simulation technique and who is

completely conversant with all of the complexities of the METRO-APEX simulation.

In addition to staffing, the other major consideration to using METRO-APEX is the cost. Although the computer programs are distributed free of charge, the manuals are expensive and the computer operational expenses can be very high. (CPU time and computer paper and printing costs can run from $120 to $200 per cycle depending on the number of runs required to correct coding and keypunch errors.)

Obviously, a professor cannot lightly and frivolously undertake the production of a simulation of the magnitude of METRO-APEX. It requires a great deal of time and effort on the part of many people, especially the G.O.D. and the computer person. On the other hand, the flexibility and complexity of this simulation are the factors which make METRO-APEX so exciting, instructional and effective as a teaching technique. Excerpts from the written evaluations required of each student and from a report prepared by David Rosen speak for themselves:

### Excerpts from Student Evaluations

"I am a history major and all you do is go to class and listen to lectures—for a change here was a class that dealt with reality and concrete problems."

"Participation in the simulation was very interesting. I learned a great deal about budgets, financing of government projects, interest group influence. . . the simulation clarified topics studied in Dr. Rosen's environmental politics course."

"This course definitely ranks among the best I've ever had."

"It was a rewarding, interesting and instructive course. I learned what it was like to be in a decision-making role; the factors, influences, considerations, priorities and benefits. I learned a great deal about myself and about human nature. If it runs next year, I would personally enjoy participating as a role advisor."

"The simulation was extremely interesting. What a challenge! A totally new type of learning experience. . . I think everyone should take this course."

"This simulation is one of the most interesting courses I have taken. I really saw how the institutional imperative works. It's such a change of pace from the traditional classroom and lecture."

"The simulation made me look forward to Tuesday nights. It was fun. I enjoyed it, but most importantly, it was of great educational value in two ways. One, it taught me the lesson of working with people, and two, it showed a realistic, even if controlled, political system. I was glad to be part of it. It will always be part of me."

"The one main thing I learned was that there exists a vast gap between the idealism of texts and classrooms and the 'reality' of a complex simulation, like METRO-APEX. In a class it all sounds so easy to balance a budget, assign priorities, clean up the environment. There is no doubt in my mind that of all my classes this is the best I have seen."

### Excerpts from a Summation Report by David Rosen

"The overall assessments of the course are overwhelmingly positive. According to the evaluations no students found the course to be uninteresting or useless and a number regarded it as the best course they have ever taken.

"Many students felt that the simulation related directly to their work in other courses.

"A number of students said that the simulation increased their interest in their 'real world' equivalents and found they were paying more attention to local politics.

"Many students asked for more theory so that they could better understand what had happened in the simulation. To any of you who have attempted to teach highly theoretical material to undergraduates, it will be clear how remarkable it is for students to ask for theory.

"Many students indicated that the simulation would be more valuable if it were supplemented with lectures and readings and made a full three-credit course."

## Conclusion

Although this writer does not advocate that standard lecture presentations be completely discarded in favor of simulations, she does suggest that gaming and simulation techniques have a place in higher education. The computer provides the power to make the model sufficiently complex so that the simulation does not become a trivial exercise. The use of the computer also frees the students and the professor from the drudgery of endless, time-consuming calculations, and allows them to focus their attention on the theories and concepts involved. If the simulation is well-designed, the computer programs are thoroughly tested and debugged and sufficient funds and competent staff are available, then institutions of higher education can offer courses which combine theoretical concepts with the equivalent of "on-the-job training," and thus provide their students with a superb learning experience.

## Note

Additional information on gaming and simulation may be obtained from sources two and three listed in the bibliography.

## Bibliography

1. COMEX Projects. METRO-APEX 1974 University of Southern California at Los Angeles (game manuals, all volumes).
2. *The Guide to Simulations/Games.* Robert E. Horn, Editor. Didactic Systems, Inc., P.O. Box 457, Cranford, N.J. 07016.
3. NASAGA (North American Simulation and Gaming Association) c/o COMEX, 3601 South Flower Street, Los Angeles, California 90007.

# From Manos to Matrices: Archaeological Classification and Computer-Assisted Cluster Analysis

**Douglas Caulkins**
*Department of Anthropology*
*Grinnell College*
*Grinnell, Iowa 50112*
*(515) 236-6181 Ext. 663*

## Introduction

Few topics in archaeology fascinate the professional more or interest the undergraduate less than the problem of classification and the construction of typologies. Since the classification of artifacts and other units is the most fundamental and time-consuming step in archaeological interpretation, we should not be surprised by the professionals' preoccupation with these problems. As Hodson notes:

> Whether an intuitive or a numerical approach is followed [in classification], and whether the resulting clusters ("types," "complexes") are regarded primarily as an arbitrary summary of data, or as a direct reflection of significant patterns of human behavior, it is clear that no interpretation of archaeological finds can be attempted until this initial stage of organization has been completed.[1]

Yet undergraduates tend to see the discussion of these problems as an esoteric and boring prelude to the real substance of courses in archaeology. This paper describes an attempt to cope with this pedagogical problem in an experimental course in archaeological interpretation at Grinnell College. A module in this course was designed to introduce students to some of the methodological issues in archaeological classification and to demonstrate some simple uses of the computer in classifying with a cluster analysis technique. This module is especially suited to the needs of small colleges which own modest archaeological collections but have access to an interactive time-sharing computer system. No previous familiarity with computer use is assumed on the part of students.

## Procedure

The module on archaeological classification and typology consisted of four phases:

*Phase 1.* Students read and discussed a set of papers on the type concept and the uses of taxonomy in archaeology. The readings were selected to represent several contrasting points of view within the discipline.[2]

*Phase 2.* The class was divided into teams, each with two or three students, for the purpose of a laboratory exercise in sorting and classifying artifacts. While instructions were deliberately minimal, we admonished students to apply what they read in the first phase. We reminded them of the distinction between ***identification***, or placing objects into already existing groups, and ***classification***, which requires the student to "create his own primary groupings, to sort all of the unknowns into new divisions."[3] The unknown artifacts confronting the students in this case were forty-three manos, or hand-held grinding stones, randomly arranged on a laboratory table. The teams were told nothing about the functions of the artifacts, the kinds of attributes which should be used to classify them, or the number of artifact types which should be generated. The students were instructed to keep a careful record of the strategy or procedure which they evolve for sorting and classifying the manos. Furthermore, they were asked to note the specific ways in which their work had been guided by the readings in phase one. We emphasized that there were no right or wrong answers to this exercise.

*Phase 3.* All teams met as a class to discuss and compare their results. As expected, they produced widely differing classifications, arrived at by different procedures.[4] The number of types generated by the teams ranged from as few as five to as many as twelve. As also expected, the readings in phase one had little apparent influence on the students' laboratory work in phase two, although some students relied upon Spaulding's classic definition of artifact type as "a group of artifacts exhbiting a consistent assemblage of attributes."[5]

Most teams first attempted an intuitive and holistic classification, but this effort was frustrated, as we had intended, by the sheer number of artifacts covering the laboratory table. Most teams abandoned this unsystematic holistic strategy and evolved a specific set of attributes to measure for each artifact. After compiling measurements on the artifacts, each team had to invent for themselves one of several systematic classification procedures. As each team described their procedure, we demonstrated how it corresponded to one of the search strategies in cluster analysis (single-linkage agglomerative, etc.), many of which are programmed for computer use.[6]

Finally, the students were told about the function and cultural context of the unidentified artifacts which they had been handling. The appropriate section of the archaeological site report was read in class and the students learned for the first time which attributes the archaeologist had chosen in classifying the manos.[7]

In phase three the students became acquainted with some of the practical problems of artifact classification and were ready to appreciate the potential contributions of the computer to this process.

*Phase 4.* In this phase the teams returned to the laboratory to classify another set of manos, this time with the aid of CLUSTR, an interactive cluster analysis program written in BASIC-PLUS, Digital Equipment Corporation's enriched dialect of BASIC.[8] The sample of artifacts was smaller this time, consisting of only twelve manos. The size of the sample was reduced for two reasons: first, to enable the students to understand precisely how the data were manipulated in the cluster analysis routine, and second, to make less burdensome the hand calculations necessary to create a data matrix for the program.[9]

As in the phase two experiment, the teams were allowed freedom of choice in selecting attributes to be measured. Teams chose as few as five and as many as ten attributes. Unlike the earlier experiment, this one included explicit directions for constructing a data matrix of N artifacts by T attributes, and for computing Jaccard's coefficient of similarity for all pairs of artifacts (see Appendix 1). An N by N matrix of coefficients of similarity is the input for CLUSTR.

After a brief demonstration of log-in procedures at a teletype terminal and thorough study of the instructions for running CLUSTR, the students were able to enter their data and run the analysis.

CLUSTR employs a polythetic divisive technique which begins with a population of N objects and partitions the population into two clusters. These clusters are in turn divided into successively more homogeneous clusters. The program allows for one element overlap between clusters.

After completing the cluster analysis, each team was instructed to take its hard copy output back to the laboratory and arrange the manos in groups corresponding to the clusters computed by CLUSTR. Most teams reported that visual inspection alone convinced them of the success of CLUSTR in creating satisfactory groupings or types of manos. The groupings of manos on the laboratory table looked right. Still, as we discovered in class discussion, different teams arrived at somewhat different types. Since the clustering technique was the same in all cases, these inconsistencies could only have resulted from differences in the teams' initial selection and measurement of attributes. Our discussion of these disparate results convinced the students of the cogency of Hill and Evans' assertion that there are empirical properties in a collection of archaeological materials, and quite often non-random clusters can be demonstrated to exist. Yet. . . it is the archaeologist who selects the variabies or attributes what he wants to consider as important in the clustering process (whether it be an intuitive or statistical process).[10]

In at least one case a team selected the wrong attributes (or too few) for CLUSTR to partition a set of artifacts which the students would have divided into subtypes on the basis of holistic visual evidence. Thus, the importance of the initial selection of artifact attributes, as well as the importance of the selection of a strategy for clustering, became vividly apparent.

## Evaluation

Students unanimously recommended that the module on classification and typology be included in future versions of the course. There was general agreement that:

a) The laboratory work contributed to an understanding of the topic which would not have emerged from the readings and lectures alone.

b) The order of the classification exercises, first without computer assistance and then with, was pedagogically very effective. Attempting to cope with the ambiguity of the phase two exercise was an important part of the learning experience, as it forced the students to re-think the apparently simple problem of classification. Neither of the two laboratory exercises should be omitted. All four phases were necessary.

Several students suggested that this module should be expanded to include cluster analysis on additional sets of artifacts.

Some students remarked that the work with CLUSTR removed their earlier prejudice against computers. The interactive program was easy to work with once the user had seen a trial run. This brief experience as computer users enabled students to better comprehend readings dealing with other computer applications in archaeology. Several humanities students even expressed a new interest in enrolling in math and computer science courses.

## Conclusion

This paper is premised on the notion that the best way for students to gain an appreciation of archaeological classification is to replicate the process, at least in some minor way. The module was a good vehicle for the examination of controversies over the type concept and the variety of classification strategies employed in archaeology. Students developed a lively sense of the fundamental importance of these issues.

A number of universities have archaeological data files, stored on cards or tape, which are readily available for analysis in standard statistical packages. While these data files may be very useful for some teaching purposes, they would not meet the needs which the author has addressed here. The student evaluations emphasize the importance of direct access to both the artifacts themselves and the computer terminal. The use of data files alone would not be as effective in illuminating the classification process. By working back and forth between the artifacts and the computer terminal, with an understanding of each transformation of the data, undergraduates are much more likely to grasp the complexities and significance of archaeological classification than they would be through standard classroom approaches to this topic.

## Acknowledgments

I wish to thank the staff of the Grinnell College Computer Services for assistance and unfailing cooperation. I am particularly indebted to Tom Moberg, coordinator of academic computing, and David Renaud, programmer/analyst. Dean Waldo S. Walker's support for this project is gratefully acknowledged. Ralph A. Luebben generously supplied the artifacts used in this instructional module.

## Appendix 1

## Preparing Data for Cluster Analysis

1. Select a population of artifacts.

2. Define a set of artifact attributes which may be coded with binary values (presence-absence). For example, the attribute "fine grain" could be coded as 1 if present and 0 if absent in a lithic artifact.

3. Construct an N x T matrix whose rows represent artifacts and whose columns represent attributes. Code each artifact and enter a value for each cell in the matrix.

Attributes

| Artifacts | A | B | C ........T |
|---|---|---|---|
| 1 | | | |
| 2 | | | |
| 3 | | | |
| . | | | |
| . | | | |
| . | | | |
| N | | | |

4. For each pair of artifacts, calculate the values for cells a, b, c, and d below:

Artifact 2

| | | 1 | 0 |
|---|---|---|---|
| | 1 | a | b |
| Artifact 1 | 0 | c | d |

Cell a represents the number of instances in which an attribute was present in both artifacts 1 and 2. Cell b represents the number of instances in which an attribute was present in artifact 1 but not in artifact 2. Cell c shows the number of attributes present in artifact 2 but not in artifact 1. Finally, cell d shows the number of attributes absent from both artifacts.

5. Using the information from the tables computed in step four, calculate Jaccard's coefficient of similarity $S_j$ for each pair of artifacts:

$$S_j = \frac{a}{a+b+c}$$

This coefficient of similarity indicates the degree of similarity of any pair of artifacts. Note that d, the absence of attributes in both artifacts, is disregarded in the coefficient. The absence of wings on fish and humans does not make them particularly similar.

6. After calculating the coefficient of similarity for each pair of artifacts, construct an N x N matrix:

Artifact

| Artifact | 1 | 2 | 3 | 4 | 5 | 6 ... N |
|---|---|---|---|---|---|---|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| . | | | | | | |
| . | | | | | | |
| . | | | | | | |
| N | | | | | | |

In the cell in row 1, column 2, write the coefficient of similarity for the pair of artifacts 1 and 2. In the cell for row 3, column 4, write the coefficient of similarity for the pair of artifacts 3 and 4. In the cells on the diagonal (1,1), (2,2), (3,3), (N,N), write the value 1 which is the maximum value for the coefficient of similarity. An artifact is, of course, maximally similar to itself! Note that the triangle above the diagonal is a mirror image of the triangle below the diagonal.

This similarity matrix can be entered in the CLUSTR program.

## Notes

1. F. R. Hodson, "Cluster analysis and archaeology: some new developments and applications," *World Archaeology*, 1 (3), 1970, p. 299.

2. James Deetz (ed.), 1971, *Man's Imprint from the Past: Readings in the Methods of Archaeology*. Little, Brown and Company, New York.

3. David Hurst Thomas, 1974, *Predicting the Past: An Introduction to Anthropological Archaeology*. Holt, Rinehart and Winston, New York, p. 6.

4. When several professional archaeologists independently classified 30 bronze brooches, the results were similarly divergent. "Widely different groupings and orderings were suggested." See J. E. Doran and F. R. Hodson, 1975, *Mathematics and Computers in Archaeology*. Harvard University Press, Cambridge, p. 223.

5. Albert C. Spaulding, "Statistical techniques for the discovery of artifact types," in James Deetz (ed.), 1971, *Man's Imprint from the Past: Readings in the Methods of Archaeology*, Little, Brown & Company, New York, p. 43.

6. Broadly defined, cluster analysis is a set of strategies for sorting observations into groups so that the degree of similarity is high among members of the same group and low between members of different groups. For a discussion of various clustering techniques see Michael R. Anderberg, 1973, *Cluster Analysis for Applications*, Academic Press, New York.

7. R. A. Luebben, "Artifacts from the Ismay Sites," Unpublished manuscript, Grinnell College.

8. The algorithm used in CLUSTR was sketched briefly by Rudo Niemeijer in "Some applications of the notion of density," in J. Boissevain and J. C. Mitchell, 1973, *Network Analysis: Studies in Human Interaction*, Mouton, The Hague, pp. 45-64. The algorithm was developed more completely and programmed in BASIC-PLUS by Tom Moberg, coordinator of academic computing, and David Renaud, programmer analyst, Grinnell College Computer Services.

9. A BASIC-PLUS program has been written to construct a similarity matrix directly from the raw data matrix, using one of several similarity coefficients. Both CLUSTR and the program to set up a similarity matrix for input to CLUSTR should run on any DEC machine using BASIC-PLUS. For further information about availability of these programs, contact Tom Moberg or Doug Caulkins, Grinnell College.

10. J. N. Hill and R. K. Evans, "A model for classification and typology," in David L. Clarke (ed.), 1972, *Models in Archaeology*, Methuen & Co., London, p. 262.

# Experimental Simulation in Introductory Sociology: Modifying EXPER-SIM to Meet the Needs of Beginners

**George H. Conklin**
*Department of Anthropology and Sociology*
*Sweet Briar College*
*Sweet Briar, Virginia 24595*
*(804) 381-5455 or 847-0442*

## Introduction

The most commonly used method of college-level instruction, especially for introductory courses, is the lecture. The usual lecture is prepared by a professor who has organized ideas, concepts and conclusions in advance and in an hour long period typically makes an opening statement, presents evidence on the subject at hand, and finally makes a conclusion. If the lecture is large, students have little or no opportunity to ask questions and must listen to an uninterrupted presentation. If the class is small, students usually feel free to ask a limited number of narrowly-focused questions designed primarily to clarify issues raised during the lecture by the professor. In either case, the emphasis is on the transmission of facts and conclusions arrived at by others at some time in the past.

Although the physical sciences usually are taught at the introductory level in large lecture courses, as a group the physical sciences have traditionally supplemented lectures with demonstrations designed to show how conclusions can be made by the experimental method. To further reinforce lectures, the natural sciences also require laboratory participation by students where experimental enquiry is taught through replication of common experiments.

Main (1972:245) has argued that the current undergraduate social science curriculum has drifted away from teaching students the fundamentals of the experimental method and has settled instead on instructing students in a set of relatively fixed concepts to be memorized. Students in the social sciences are taught theories advanced by others or perhaps even the results of some research report applying theories. But unlike the laboratory sections in a course such as chemistry, the social-science student is given no direct feeling for what it is like to try to prove or disprove a theory by direct examination of the evidence.

In teaching introductory sociology, the author has noted typical questions by students which show the lack of experience in trying to carefully weigh evidence in making conclusions about human behavior. Students often ask complex questions and then feel cheated if a simple answer is not forthcoming. For example, in discussing the role of nature and nurture in determining human behavior, students would typically ask which was the "real" cause and feel the instructor was avoiding the question if the answer given did not enable them to write one or the other down for their notes. Often a tendency to overestimate the role of the computer would surface when students would ask "in principle" questions such as, "Well, if you had a big enough computer, could not all questions be settled?" Obviously what was needed even at the introductory level was a sociological equivalent of the laboratory experience offered by the physical sciences.

## Experiment Simulation on the Computer

In searching for a way to introduce the social science student to the equivalent of the laboratory situation used in the physical sciences at a reasonable cost, the author decided to use a set of existing programs developed by Main (1974, 1975; also Thurmond and Cromer 1975) called EXPER-SIM, short for experiment simulation. EXPER-SIM is a group of related programs which simulate an experiment by letting the student develop a hypothesis and then test that hypothesis by exploring the ways in which variables interact. Typically a student selects values for five or six variables and the computer then calculates an outcome and adds error to the calculated answer. Students then must average together several trials to obtain an unbiased answer. Originally conceived as a system of teaching programs for the majors' methods course in psychology at the University of Michigan, EXPER-SIM was evaluated at the University of Michigan by Nussloch (*et al.* 1974) and found to be significantly better in teaching hypothesis testing than conventional lectures and assigned homework. As with the physical sciences, there seems to be a built-in advantage in letting students see for themselves not only general conclusions but also in personally delving into some of the complexities of analysis.

## Problems Faced by Introductory Students With EXPER-SIM

Introducing EXPER-SIM into an introductory sociology course proved far more difficult than at first anticipated. The first effort was made in the fall of 1975 when the SCHIZ model, originally developed by Malin (1974), was made available to introductory students. The SCHIZ model used was a Honeywell FORTRAN translation of the model as it appears on the Louisville Experiment Simulation System (LESS) version of EXPER-SIM. The experiment was discussed in class as a simulation of the different effects of the ways in which individuals are socialized. Students were given a hand-out consisting of the possible causes of schizophrenia (Malin 1974:57-60) and a decision sheet explaining the variables and the values each variable could take. Since the model used was identical to the LESS version of SCHIZ, at that time SCHIZ merely asked the student if she wanted to be prompted (y or n) and if so, typed out the names of each variable (as in the Appendix). If not, the program asked for seven parameters followed by commas. No other instructions were offered or available to the student.

In theory using SCHIZ seemed simple enough. But in fact, introductory students found the unmodified EXPER-SIM program more confusing than expected. It became another burden to be finished and thus only poorly taught the principles of the experimental method. Even though the students had been instructed otherwise, they would continually: (1) change two variables at a time making it impossible to tell which change had produced a certain effect, (2) spend hours trying to work all combinations, only to give up in frustration, or, (3) try to save time by doing only one replication, thus failing to average out the simulated measurement error.

In writing up the reports, the freshmen faced a problem common to graduate students: how to structure an answer. An even larger problem was the sense of panic developed by about a quarter of the class which felt that the machine was pushing them and not giving them sufficient chance to think. While the very purpose of the experiment simulation was to show that every possibility could not be tried, some who had not actually tried out all combinations wrote in their reports that they felt fewer trials should be made so more and different combinations could be tried out. Lastly, report writing reflected the fact that the students, at least at Sweet Briar College, tended to forget to take the decision sheet with them to the terminal room and thus were confused when the program itself did not contain a full set of instructions. Rather than help the course, the unmodified SCHIZ model became a detriment.

## Modifying EXPER-SIM for Freshmen

My initial feeling that to use EXPER-SIM at the introductory level it would be necessary to modify the printed instructions and the handouts proved to be false. What was necessary was to modify the program itself. More specifically, it turned out that to make SCHIZ and experiment simulation meaningful for a general survey introductory sociology course, the computer program itself would have to be self-contained. Students expected to use the computer much as they would the textbook—to explain something that they did not understand. The command "Enter 7 parameters followed by commas" merely confused rather than instructed the beginner.

The modifications which made EXPER-SIM successful for the freshman can best be illustrated by examining the output in the Appendix. For sake of brevity, the modifications and what was accomplished by each are listed below:

1. The title and text: To tie the program to the substantive issues raised in the course, several basic highlights on the debate over the relative strengths of nature and nurture on the individual are mentioned. This places the experiment in perspective before facing the student with any decisions. It helps make the student feel at home before requiring any active intervention.

2. The experiment: This paragraph introduces the problem of measurement and measurement error. It also reminds the student of the importance of changing only one variable at a time in order to compare two groups of people. Since the purpose of the entire exercise is to teach the experimental method, this reminder is very important for students to keep in mind.

3. The choices to be made: To overcome the problem of the student who forgets the decision sheet, and especially to encourage a careful reading of the options, the decision sheet is included in the text, not as a handout in class.

4. The record of decisions and trials: The record of decisions and a place to write down the trial scores solved the problem of the student who literally lost her findings on a twenty-foot roll of output. By forcing the student to copy each answer into the correct column as it is computed, attention is drawn to the actual values which previously were often ignored until sometime later. After typing out the blank tables the computer pauses for twenty seconds to enable students to tear off the output and get ready to proceed.

5. Instructions: The instructions remind each student of what she needs to know in order to complete the experiment. At the end of the instructions, the machine pauses until instructed to proceed. This pause is extremely important for those who have little confidence in their abilities on the computer. It gives beginners the time to study the instructions and to make some decisions on what to do next. While the student who is confident simply proceeds without delay, poor students gain confidence by being told "you are in control."

6. The Louisville Simulator: The Louisville Experiment Simulator was modified in four ways. First, the student is always asked for each variable by name. Secondly, whatever decisions are chosen are retained for ten trials. This modification cut down dramatically on errors while forcing each student to conduct ten trials. Thirdly, to allow time to write down each test score, the return button must be pushed for the program to continue. Lastly, the random number generator was slightly modified to permit a truly random seeding value for each experiment.

7. Calculations of means and medians: As an aid to teaching the meaning of several measures of central tendency, students are permitted to call in a specially modified library routine to compute the means and medians of their test values and, by placing the scores in rank order, the program also lets the student see if a mode is present. Because of the modifications to the program, students need not be careful not to put a comma at the end of each data line, another common error. If a value is left out, the machine checks to see if all 10 values have been entered; if not, it asks for more data.

8. Error corrections: If an error has been made in entering data, a student is permitted to return to the start of the program which calculates the measures of central tendency.

9. Instructions for the homework: Lastly, students are given brief instructions on hand-

ing in the homework. Since the tables upon which the students enter their findings are exactly eleven inches long, the results fit neatly in a folder. Because many students had seen "Sybil" on television or had read the book (which was on the best seller list during the fall of 1976), it was suggested that their "scientific" findings might be compared with a literary interpretation by a popular author.

### The Results

The above modifications turned EXPER-SIM into a highly useful teaching tool for a general survey course dealing with students who had had little experience on the computer. Although time consuming to enter all the modifications, the most common problems faced by the introductory student seem to have been met. The proportion of papers turned in where the student had been able to make useful conclusions rose from 62 percent without the modifications to nearly 95 percent with the more complete instructions.

More importantly, the two students who run the aid desk and who were assigned to the introductory sociology sections reported that while in the fall of 1975 "many" students had asked for help with the unmodified program, in the fall of 1976 with the modified SCHIZ model only a small fraction of those doing the assignment needed aid. The most common problem remains careless reading of instructions. One student, however, read the number "100" as "1" because she felt the "00" were zeroes that the teletype had "crossed out" as errors. There seems to be no way to accurately predict this type of problem.

### Conclusions

Introductory courses in the physical sciences use experiments in the laboratory as an aid in instruction. Introductory social science courses could also benefit from introducing experiment simulation into the curriculum. Properly used the computer should enable the instructor to include the benefits of the laboratory at a reasonable cost into a pre-existing course. At present, the author combines EXPER-SIM with a program for introductory sociology (PROFIS) developed earlier (Conklin 1976). Together, both programs enable inclusion of many techniques of analysis without otherwise detracting from a crowded curriculum. Experience has shown, however, that unless the special needs of the beginning student are taken into account, the computer can also introduce confusion into an otherwise well-organized class.

### A Technical Note

The modifications of EXPER-SIM described here were carried out in BASIC on a Honeywell 1640 which allows no string variables or programs of over 150 lines in BASIC. The decision to keep the text in BASIC and the simulator in FORTRAN was made because disk space is at a premium while compute time is freely available. While most small colleges like Sweet Briar College probably have small computers, sometimes small computers also lack a billing system which enables their use at will by instructors. Such is the case at Sweet Briar.

The SCHIZ model is one of the models in common with both the Michigan version of EXPER-SIM (MESS) and the Louisville version (LESS). While frequent chaining and the piecework nature of the modifications may sound crude to those familiar with larger computer systems, small college computers often lack disk space, forcing storage of longer programs in various accounts. Further, since students do not see these problems in their printouts, they do not affect the outcome in terms of learning efficiency. At regular ASR 33 teletype speed of ten characters per second, the average student spends fifty-five minutes on the homework assignment described here.

### References

Conklin, George H. (1976) "Using the Computer in Introductory Sociology: A Solution to the 'Methodology Problem'." *Teaching Sociology* 4 (October):83-95.

Main, Dana B. (1972) "Toward a Future-Oriented Curriculum." *American Psychologist* 27 (March):245-248.

________ . (1974) *EXPER-SIM on Michigan Experimental Simulation Supervisor.* Xerox College Publishing.

________. (1975) "An Overview of the EXPER-SIM (Experiment Simulation) Pedagogy." in Lecarme, O. and Lewis, R. (eds). *Computers in Education, Part II.* New York: Elsevier Publishing Company.

Nussloch, Richard H., Richard J. Kaplan, and Dana B. Main (1974) "Causal Critiques or Active Arguments?—A Comparison of the Use of Computer Data Generators and Journal Article Criticism in the Science Classroom." Presented at the Fifth Conference on Computers in the Undergraduate Curricula. Washington State University. Reference to authors' reprint.

Malin, David (1974) "The Etiology of Schizophrenia (SCHIZ)" in Main, Dana, ***EXPER-SIM on the Michigan Experimental Simulation Supervisor.*** Xerox College Publishing 1974, pp. 57-61.

Thurmond, John B. and Arthur O. Cromer (1975) "Models and Modeling with Louisville Experiment Simulation System (LESS)." ***Behavior Research and Instrumentation*** 7: 229-232.

## APPENDIX (Conklin)

### THE FAMILY AND SOCIAL BEHAVIOR

IN MOST IF NOT ALL SOCIETIES OF THE WORLD, CHILDREN ARE BORN INTO A FAMILY AND IT IS THE FAMILY WHICH HAS THE FIRST CHANCE TO SOCIALIZE AND INFLUENCE THE CHILD. EFFORTS TO SOCIALIZE CHILDREN BEGIN IMMEDIATELY AFTER BIRTH. THE BABY WHICH SLEEPS ALL DAY AND CRIES ALL NIGHT IS STRAIGHTENED OUT QUICKLY BY SLEEPY PARENTS. OUR CULTURE EVENTUALLY SOCIALIZES THE CHILD TO GET HUNGRY THREE TIMES A DAY AT BREAKFAST, LUNCH AND SUPPER, WHILE OTHER CULTURES BELIEVE THAT THE HUMAN ANIMAL NATURALLY GETS HUNGRY JUST TWICE A DAY.

ON A MORE COMPLEX LEVEL, CHILDREN ARE SOCIALIZED INTO ADULT ROLES AS THEY GROW. SOME SOCIOLOGISTS SUCH AS WESTON LA BARRE HAVE ARGUED THAT BOTH NATURE AND NURTURE HAVE PLAYED A PART IN THE FORMATION OF HUMAN SOCIETY TAKEN AS A WHOLE. LA BARRE THINKS THAT THE EVOLUTION OF THE HUMAN FAMILY DEPENDED ON WHAT HE CALLS THE YEAR ROUND SEXUALITY OF WOMAN. IN EARLIER TIMES IT WAS NECESSARY FOR MEN TO HELP RAISE THE CHILDREN BECAUSE HUMAN INFANTS ARE BORN SO COMPLETELY HELPLESS THAT THEY REQUIRE INTENSE CARE IN ORDER TO SURVIVE. FOR LA BARRE, BOTH NATURE AND NURTURE ARE IMPORTANT IN INFLUENCING THE HUMAN FAMILY AND IN, IN HIS WORDS, GETTING THE FATHER TO STAY HOME.

WHEN SOCIALIZATION IS SUCCESSFUL AND A CHILD GROWS UP TO BE A REASONABLY SUCCESSFUL ADULT, FEW PEOPLE TAKE MUCH INTEREST IN THE DETAILS OF THE SOCIALIZATION PROCESS. BUT WHEN SOMETHING GOES WRONG AND A CHILD BECOMES DEVIANT OR MENTALLY ILL, THE QUESTION IS OFTEN RAISED, WHAT WENT WRONG? IS A PERSONS INABILITY TO MAKE DECISONS OR FUNCTION AS AN ADULT PART OF HIS OR HER EARLIER TRAINING? WOULD THE INDIVIDUAL HAVE TURNED OUT BETTER IF SOCIALIZED BY A BETTER OR MORE COMPETENT SET OF PARENTS? THESE AND MANY OTHER QUESTIONS CAN BE RAISED UNDER SUCH CIRCUMSTANCES.

SCHIZOPHRENIA IS AN EXAMPLE OF DEVIANT BEHAVIOR WHERE THE ETIOLOGY (ORIGIN) OF THE DISEASE IS NOT AGREED UPON. SOME AUTHORS TRACE ITS ORIGIN TO THE DOUBLE BIND WHERE A PERSONS PARENTS WOULD ASK A CHILD NOT TO DO SOMETHING AND THEN WOULD PUT THE CHILD DOWN FOR DOING THE OPPOSITE...A SORT OF HEADS I WIN TAILS YOU LOSE SITUATION. THE RESULT IS THAT A CHILD GROWS UP UNABLE TO FUNCTION AS AN ADULT.

RATHER THAN SOCIALIZATION OR NURTURE AS A SOURCE OF SCHIZOPHRENIA, SOME AUTHORS SAY THAT NATURE PLAYS A CRUEL TRICK ON SOME UNFORTUNATE PERSONS. THERE IS SOME EVIDENCE TO SUGGEST THAT SCHIZOPHRENIA IS CAUSED BY A PERSON BEING UNABLE TO PROCESS CERTAIN CHEMICALS PROPERLY AND THUS A SCHIZOPHRENIC IS ACTUALLY INTOXICATED BY CHEMICALS SIMILAR TO LSD. A GENETIC MISTAKE CAUSED BY NATURE IS THUS SEEN AS A CAUSE OF DEVIANT BEHAVIOR, NOT IMPERFECT SOCIALIZATION BY MISGUIDED PARENTS IN THE FAMILY.

## THE EXPERIMENT

HOW WOULD A REASONABLE PERSON GO ABOUT DETERMINING THE RELATIVE IMPORTANCE OF SOCIALIZATION (NURTURE) AND NATURE IN DETERMINING DEVIANT BEHAVIOR?
ONE WAY IS TO CONDUCT AN EXPERIMENT TO EXAMINE THE FAMILIES OF THOSE WHO LATER BECAME MENTALLY ILL. YOU MIGHT ASK YOURSELF WHAT WAS DIFFERENT IN THE FAMILIES OF THOSE WHO PRODUCE MENTALLY ILL CHILDREN AS COMPARED TO THOSE FAMILIES WHERE THE CHILDREN ARE NORMAL. TAKE THE CASE OF IDENTICAL TWINS. THERE ARE CASES ON RECORD WHERE IDENTICAL TWINS HAVE BEEN SEPARATED AT BIRTH AND RAISED BY DIFFERENT FAMILIES. IF BIOLOGY WERE THE CAUSE, THEN YOU COULD LOOK AT THE RATES OF MENTAL ILLNESS AMONG IDENTICAL TWINS RAISED BY DIFFERENT FAMILIES TO SEE HOW SIMILAR (OR DIFFERENTLY) THE TWO GROUPS TURNED OUT.

OTHERS BEFORE YOU HAVE EXAMINED THE EFFECTS OF NATURE AND NURTURE ON CHILDREN SOCIALIZED IN DIFFERENT TYPES OF FAMILIES. THIS TEACHING PROGRAM ENABLES YOU TO EXAMINE SOME OF THE RESULTS OF EXISTING RESEARCH.

IN THE REAL WORLD, MEASURING THE RATE OF MENTAL ILLNESS AMONG VARIOUS GROUPS IS EASIER SAID THAN DONE. SOMETIMES THE EXPERTS DISAGREE ON THE DIAGNOSIS, OR THEY MAY DISAGREE HOW SEVERE THE ILLNESS IS. ALSO, THERE IS RANDOM ERROR INVOLVED IN MEASUREMENT, SO TRUSTING ONE STUDY IS NOT GOOD ENOUGH. TO OVERCOME ERROR, MOST STUDENTS OF SOCIETY ASSUME IT IS BEST TO TAKE THE AVERAGE OF SEVERAL DIFFERENT TRIALS OR MEASUREMENT EFFORTS AND AVERAGE THE RESULTS.

```
RUN

    THE CHOICES YOU CAN MAKE IN YOUR MEASUREMENT EFFORTS
ARE AS FOLLOWS:
 1. NUMBER.   HOW MANY PEOPLE DO YOU WISH TO MEASURE?  USE
              100. THE MAXIMUM PERMITTED RANGE IS 20 TO 225.
 2. INDEX.    DIAGNOSIS OF INDEX SUBJECT.  1 = SCHIZOPHRENIC
                                           2 = NONSCHIZOPHRENIC.
 3. RELATION. WHAT IS THE RELATIONSHIP OF YOUR GROUP TO THE
              INDEX SUBJECT.  FOR EXAMPLE, IF YOU CHOOSE 8,
              AND NUMBER = 100 AND INDEX = 2, YOU WILL BE
              EXAMINING THE PROPORTION OF THAT GROUP OF 100
              OFFSPRING TO BECOME MENTALLY ILL LATER.
              1=UNRELATED     2=AUNT/UNCLE    3=COUSIN
              4=SIBLING       5=IDENT. TWIN   6=NON-IDENT. TWIN
              7=PARENT        8=OFFSPRING
 4. ADOPT.    NATURE OF RELATIONSHIP OF YOUR GROUP TO ACTUAL
              INDEX SUBJECT...IF APPLICABLE. 1=BIOLOGICALLY
              RELATED AND NEVER SEPARATED. 2=BIOLOGICALLY
              RELATED BUT SEPARATED IN INFANCY. 3=RELATED BY
              ADOPTION IN EARLY INFANCY.
```

```
  5. DIAGNOSIS. STRICTNESS OF DIAGNOSIS.  1=SCHIZOPHRENIC
               PSYCHOSIS ONLY--A STRICT DIAGNOSIS.  2=SCHIZOID
               PERSONALITY ALSO.
  6. SEX.      1=SAME AS INDEX SUBJECT, 2=OPPOSITE.
  7. METHOD.   METHOD OF DIAGNOSIS. 1=SAME PERSON LOOKS UP INDEX
               SUBJECTS, ACTUAL SUBJECTS AND DIAGNOSES THE LATTER.
               2=DIAGNOSIS PERFORMED BY SOMEONE WHO DOES NOT KNOW
               IF INDEX SUBJECT IS SCHIZOPHRENIC OR NOT.


                      HOW TO CONDUCT YOUR EXPERIMENT

     BELOW WILL BE PRINTED OUT A DECISION SHEET.  WHEN THE
 DECISION SHEETS ARE PRINTED OUT, THE COMPUTER WILL PAUSE FOR
 15 SECONDS.  PLEASE TEAR THE SHEET OFF THE TELETYPE SO YOU CAN
 WRITE ON IT.  THEN AWAIT INSTRUCTIONS.
RUN



     RECORD OF DECISIONS: WHAT I MEASURED

 MY CHOICES FOR EXPERIMENT NUMBER:(FILL IN)

 VARIABLE ! EXPERI-  ! EXPERI-  ! EXPERI-  ! EXPERI-  !
  CHOICE  ! MENT # 1 ! MENT # 2 ! MENT # 3 ! MENT # 4 !
 ---------!----------!----------!----------!----------!
 NUMBER:  !          !          !          !          !
 ---------!----------!----------!----------!----------!
 INDEX:   !          !          !          !          !
 ---------!----------!----------!----------!----------!
 RELATION:!          !          !          !          !
 ---------!----------!----------!----------!----------!
 ADOPT:   !          !          !          !          !
 ---------!----------!----------!----------!----------!
DIAGNOSIS:!          !          !          !          !
 ---------!----------!----------!----------!----------!
 SEX:     !          !          !          !          !
 ---------!----------!----------!----------!----------!
 METHOD:  !          !          !          !          !
 ---------!----------!----------!----------!----------!


 RECORD BELOW THE INDIVIDUAL TRIAL SCORES FOR EACH EXPERIMENT:

       NUMBER OF PERSONS FOUND TO BE SCHIZOPHRENIC PER TRAIL:
                 (FILL IN)

           EXPERI-  ! EXPERI-  ! EXPERI-  ! EXPERI-  !
           MENT # 1 ! MENT # 2 ! MENT # 3 ! MENT # 4 !
           ---------!----------!----------!----------!
TRIAL #1:           !          !          !          !
--------------------!----------!----------!----------!
TRIAL #2:           !          !          !          !
--------------------!----------!----------!----------!
TRIAL #3:           !          !          !          !
--------------------!----------!----------!----------!
TRIAL #4:           !          !          !          !
--------------------!----------!----------!----------!
TRIAL #5:           !          !          !          !
--------------------!----------!----------!----------!
```

```
TRIAL #6:           !           !           !           !
--------------------!-----------!-----------!-----------!
TRIAL #7:           !           !           !           !
--------------------!-----------!-----------!-----------!
TRIAL #8:           !           !           !           !
--------------------!-----------!-----------!-----------!
TRIAL #9:           !           !           !           !
--------------------!-----------!-----------!-----------!
TRIAL #10:          !           !           !           !
--------------------!-----------!-----------!-----------!
 AVERAGE OF         !           !           !           !
 TEN VALUES:        !           !           !           !
--------------------!-----------!-----------!-----------!
 MEDIAN OF          !           !           !           !
 TEN VALUES:        !           !           !           !
--------------------!-----------!-----------!-----------!
 MODE OF            !           !           !           !
 TEN VALUES:        !           !           !           !
--------------------!-----------!-----------!-----------!




-------------------------(TEAR OFF HERE)-------------------------




RUN
              INSTRUCTIONS

     1. LOOK AT THE POSSIBLE VALUES LISTED ABOVE FOR EACH
        DECISION.

     2. MAKE A DECISION ON WHICH VALUES YOU WISH TO TEST.

     3. USING A PENCIL, WRITE THOSE VALUES DOWN IN THE COLUMN
        LABELED  EXPERIMENT # 1.

     4. AFTER YOU HAVE LISTED THE NUMBERS YOU WISH TO USE,
        PLEASE ENTER THE VALUE 1 INTO THE TELETYPE, WHICH WILL
        NOW STOP AND WAIT FOR YOU TO FINISH.

     5. THE EXPERIMENT SIMULATOR WILL ASK YOU FOR YOUR VALUES,
        WHICH YOU SHOULD THEN ENTER.  TRIAL # 1 WILL THEN BE
        DONE AND THE ANSWER PRINTED.  WRITE DOWN THE FIRST
        ANSWER IN THE SPACE PROVIDED ABOVE UNDER EXPERIMENT # 1

     6. WHEN YOU ARE READY FOR THE SECOND TRIAL, PUSH RETURN AND
        THE SECOND TRIAL WILL BEGIN.  AFTER TEN TRIALS, YOU WILL
        BE ASKED TO CHANGE YOUR CRITERIA.
 WHEN READY TO RESUME, TYPE 1 AND PUSH RETURN
!1

 LOUISVILLE EXPERIMENT SIMULATION SYSTEM
 NATURE OR NURTURE SIMULATION
 IN REFERENCE TO THE DECISION SHEET ABOVE, PLEASE
  ENTER THE CODE NUMBERS FOR WHAT YOU WANT TO STUDY.
NUMBER  =?! 100

INDEX   =?! 1

RELATION=?! 5

ADOPTION=?! 1

DIAGNOS =?! 2

SEX     =?! 1

METHOD  =?! 2

  24 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  32 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  28 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  30 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  29 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  30 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  30 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  24 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  30 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.
 PRESS RETURN WHEN YOU ARE READY FOR THE NEXT TRIAL!

  31 PERSONS IN THIS GROUP OF 100 WERE DIAGNOSED AS SCHIZOPHRENIC.RUN


     YOU HAVE NOW SUCCESSFULLY COMPLETED TEN TRIALS WITH ONE
 SET OF VALUES.  NOW, BEFORE PROCEEDING, BE CERTAIN TO HAVE
 ALL YOUR VALUES CORRECTLY LISTED IN YOUR TABLE.

     NEXT, AVERAGE THE TEN VALUES (ADD THEM AND DIVIDE BY TEN)
 TO GET THE BEST SINGLE MEASURE OF THE INCIDENCE OF SCHIZOPHRENIA
  IN YOUR TRIAL GROUPS.

     ALSO, FILL IN THE MEDIAN OR MIDDLE-VALUE AND THE MODE
  OR SINGLE MOST COMMON VALUE IN THE PLACES PROVIDED.
```

```
    WOULD YOU LIKE SOME HELP IN DOING THE AVERAGE?  IF.SO,
 TYPE 1.  IF NOT TYPE 2 AND DO THE AVERAGES BY HAND.

!1
RUN


    THIS PROGRAM FINDS THE MEAN AND MEDIAN OF YOUR 10 TRIAL
  SCORES FOR YOUR EXPERIMENTS.  WHEN THE ! APPEARS, ENTER
  5 OF YOUR VALUES ON ONE LINE SEPARATED BY COMMAS.  PUSH RETURN
    THEN ENTER THE SECOND SET OF 5 VALUES SEPARATED BY COMMAS
  AND PUSH RETURN.


  TYPE IN 5 TRIAL SCORES PER LINE SEPARATED BY COMMAS:
!24,32,28,30,29
O.K. PUT IN 5 MORE
!30,30,24,30,31


 THE SCORES FROM YOUR EXPERIMENT IN RANK ORDER ARE AS FOLLOWS:

    32          31          30          30          30

    30          29          28          24          24


NO. OF SCORES  THE MEAN IS  THE MEDIAN IS

    10          28.8        30


    PLEASE CHECK TO MAKE CERTAIN YOU HAVE CORRECTLY ENTERED THE
 CORRECT VALUES FROM YOUR DECISION SHEET.  IF BY CHANCE YOU HAVE
 A MISTAKE, THEN TYPE 1 AND YOU WILL AGAIN BE ABLE TO ENTER YOUR
 VALUES.  IF YOU ARE READY TO PROCEED, TYPE 2.
    IF YOU ARE DONE WITH ALL FOUR EXPERIMENTS AND WISH TO STOP,
 THEN TYPE 3 AND THE PROGRAM WILL SIGN OFF.
!3
RUN

    IF YOU HAVE NOW COMPLETED YOUR FOUR EXPERI-
 MENTS, PLEASE WRITE UP YOUR THOUGHTS ABOUT THE
 CORRELATES OF SCHIZOPHREINIA IN ONE OR TWO TYPED PAGES.
 CONSIDER THE RELATIVE FORCE OF NATURE AND THE RELATIVE FORCE OF
 NURTURE.  WHICH DO YOU FEEL INFLUENCES BEHAVIOR?  EXPLAIN WHY.
 BE CERTAIN TO ATTPATCH TO YOUR DISCUSSION THE DECISION
 SHEET WHICH YOU HAVE FILLED OUT.  IT IS NOT NECESSARY
 TO TURN IN THE REST OF THE OUTPUT.

    DID YOU SEE SYBIL ON T.V. THIS FALL?  YOU MAY MAKE REFERENCE
 TO THAT FILM IF YOU WISH.



  170 EXIT

?
```

# Effectively Teaching Undergraduate Social Research Via Computer Data Analysis

**William R. Brown**
**Ida Cook**[1]
**Charles M. Unkovic**
*Department of Sociology*
*Florida Technological University*
*Orlando, Florida 32816*
*(305) 275-2227*

### Introduction

History will no doubt record along with her most hallowed inventors the names of Charles Babbage, George Hollerinth, Howard Aiken, Konrad Zuse, and later visionary wizards who contributed importantly toward the conceptualization of what has become our modern computer era (Brier and Robinson, 1973). And rightfully so! Our space-age technology is recognized as being increasingly dependent on this electronic marvel.

The rapidly expanding usage of computer programs, especially "packaged" subprograms, has also given the social scientists capabilities that few of them dreamed possible even a decade ago. While tremendous strides have been made *technologically* with respect to efficiently handling and modifying social data, there exists still a lag in training a sufficient number of social scientists to competently use these apparati with a maximal degree of effectiveness. It is upon this latter premise that this paper is based.

A rather unique approach designed to interest and assist undergraduate sociologists in learning what has been previously regarded as "advanced" methodologies is described below. A number of innovations which have made practical the use of computers as very effective learning tools for undergraduate social scientists, especially sociology students, are also discussed.

Prior to describing these strategies involving the computer, it seems appropriate to introduce the rationale for the need for greater emphasis on undergraduate social research.

### Undergraduate "Hard Sciences" and "Social Sciences:" The Methodological Emphasis Gap

When one compares the relative emphasis given to undergraduate empirical research methods courses for some of the so-called "hard sciences" to those of the "social sciences," the methodological "gap" becomes apparent. These comparisons are easily gleaned in Table 1.

When the totals of the four research courses in the "hard" and "social" sciences are compared, 33.5 percent compared to 12.5 percent of the above total course curricula are methodologically oriented within the "hard" and the "social" sciences, respectively. Clearly more than twice as much emphasis is placed on "methods" within the "hard sciences" compared to the "social sciences," at least on the basis of the relative proportion of methods courses offered within each of the above departments.[2]

A very defendable argument could also be generated to the effect that the research methodology in the "social sciences" is in many aspects just as difficult, if not more difficult, to learn and to teach as are the methodologies in the "hard sciences." For example, human subjects often react differently when they know they are being observed (e.g., the "guinea pig" phenomenon). Furthermore, human subjects have a multitude of

**Table 1. Comparison of Percent of Research Methods Courses Offered to Total Courses Offered by Four Major Departments in "Hard Sciences" and in the "Social Sciences"[a]**

| | *Hard Sciences* | | | | *Social Sciences* | | | |
|---|---|---|---|---|---|---|---|---|
| | Biology | Botany | Chemistry | Physics | Political Science | Sociology | Communication | Psychology |
| Percent Methods Courses Offered | 27% | 34% | 36% | 37% | 6% | 11% | 12% | 21% |
| Total No. Courses Offered | 24 | 16 | 39 | 34 | 50 | 56 | 36 | 35 |

stimuli impinging upon their subjective attitudes and public behaviors. Laboratory experiments are quite artificial settings for humans, making the generalization of findings very difficult (Smith, 1975; Babbie, 1975). While the "which is the most difficult" argument could never be resolved to everyone's satisfaction, the fact remains that the methodologies necessary for quality social research *are* very complex. Yet 22 percent of the universities and 38 percent of the four-year colleges do not presently offer even a *single* course in undergraduate research methods.[3] From our own experience we are convinced that most students need additional as well as more effective training than they now generally receive if they are in fact going to become active social scientists. With so little emphasis, it is understandable but sad to note that many of our bachelor degree students cannot even adequately read their professional journals.

## The Need for Greater Emphasis on Undergraduate Social Research Methods

Let us now focus on undergraduate social research methods. Largely due to recent, sophisticated computerized statistical packages such as SPSS (Nie et al., 1975), sociology now has within its grasp the potential for contributing much more and, importantly, much better quality information that is sorely needed within both the governmental and private sectors. Given these relatively new and far more pragmatic electronic capabilities, sociologists can now work much more effectively with multivariate analysis techniques on complex sets of data. "Path" models illustrating the complex ways that variables work through and with other variables are now feasible. Not only can we study and understand complex social processes more readily, we can now more accurately evaluate and predict their magnitude and direction due to various changes in social structures of society and social organization. Furthermore, new and improved computer subprograms are being developed jointly by computer specialists, statisticians, and social researchers. In that other social sciences disciplines are not as concerned with complex sets of data (i.e., utilizing numerous variables), sociology in particular now has a very marketable product even though its full potential is just beginning to be realized. As a discipline we are on the brink of being recognized as a major contributor of useful types of information that have been lacking when important decisions have been considered. We must now alert industrial, business, and governmental sectors of our greatly improved capabilities. It's time they learn what sociologists can now do for them.

The wider utilization of sociological methodologies means not only can sociology make important contributions to society and to its social organizations but also that research oriented sociology students are going to be in far greater demand as this potential is realized. If one factor were to be singled out as largely responsible for the bright future of sociology, it would be the recent development of relatively easy to learn and use packaged computer programs which allow us to handle complex sets of data with far greater efficacy.

---

[a] These figures are approximations for it was not always possible to ascertain the precise proportion devoted to methodology of some lecture/lab type courses. Nonetheless, the comparisons are all based on clearly identifiable methodological courses for each particular major. These figures are based on the data in the *Florida Technological University 1975-76 Catalogue.*

The foregoing is not to say that social data are now, ever were, or ever will be, precise. Social data are not, never have been, and probably never will be *precise* . Nonetheless, sociological data *are useful* now, and the data will become more and more useful as we train more and better research sociologists in advanced areas of computerized, statistical technologies. Fully learning and understanding how to work with such interrelated variables requires numerous methodological and statistical courses, some of which need to be oriented around computer skills. In addition, students need a great deal of actual practice in *doing* research. It is our firm belief that we need to vastly upgrade the statistical and methodological curriculum within undergraduate sociology departments to meet the growing demand for our unique, much-needed product. While the cost of becoming a social methodologist will be higher, the caliber of our product will increase accordingly.

We should note, however, that only a relatively small proportion of sociology students usually wish to enter seriously into the social research aspects of sociology; most students have preferred substantive or theoretical interests. As Blalock (1975) pointed out, we need such a division of labor within sociology. However, methodologists need to be better trained to better carry out their specialized functions. We may do well in sociology to challenge and recruit the best (in terms of mathematically, "scientifically" inclined students) high school students to enter sociology and become social researchers. By training undergraduates with real research interests in more advance methodologies, these students can enter graduate school much better prepared to truly achieve sophistication in social research even at the master's level.

## Strategy for Teaching Undergraduate Social Research

In an effort to upgrade our FTU undergraduate program with respect to social research emphasis, we have developed two back-to-back courses, "Data Collection Methodology" and "Data Analysis." The content and rationale for each of these courses are discussed in some detail below, although we shall here focus largely on "Data Analysis" since this course is designed around the use of the computer as a skill and, even more importantly, as a tool.

## First Quarter: "Data Collection Methodology"

During the first course, students learn the basic techniques necessary to collect valid information for subsequent analysis. After all, a scientist's data must be based on valid information or data analysis is futile. As noted earlier, we cannot often measure social phenomena perfectly; however, we can and do measure social indicators so as to allow these data to provide very useful information and guidelines. Proper training in these methodologies is essential.

In addition to studying the basic methodological principles involved in collecting reliable information with care being given to learning to avoid numerous sources of error, the students study various sampling techniques and strategies, the appropriateness of available research designs, as well as the close relationship between theory and research. They also learn what mode, or combination of modes, of data collection are most appropriate for each particular research problem.

For example, the students study the advantages and disadvantages of interview schedules, survey questionnaires, laboratory and field experimental techniques, working with documents, content analysis, participant observation, and using unobtrusive data collection techniques. However, since questionnaires are by far the most commonly used (and also least expensive means) of collecting data we focus much more on this mode.

### *The Key to Learning Research: Involvement*

Fairly early in the quarter, students (usually about twenty-five per section) are asked to divide into teams of two to four compatible members and then carefully select a topic of research that they *want* to study. Example topics which students have previously selected include: "Suicidal Tendencies Among College-Age Students," "Determinants of Students' Perceptions of Police Officers," "The Interaction Between 'Severity of Punishment' and 'Certainty of Punishment' as Crime Deterrents," "Changes in Touching Taboos," "Crying and Sex Roles," "Singles' Battles in Today's World," "Predictors of Marijuana Usage and Discontinuance," "The Bar Room and Social Behavior," "Sexual Attitudes and Behaviors," and "Alternatives to Referendums at the Polls."

Once the general topic is agreed upon, each team of students must write a proposal to study this topic. The proposal entails writing an introductory section that justifies the need for information on the topic, its importance, and the broader aspects of the topic. Via a division of labor, each team then searches the professional literature for empirical and theoretical studies that relate to its area of interest. The students learn to summarize others' findings in a clear, concise manner.

From its literature search, as well as from logic, each team must formulate and justify a number of hypotheses about its topic. The students must also list all other "test" variables they can justify for inclusion in their study. Learning to master these conceptual skills is crucial in order to provide direction in the subsequent data collection and data analysis procedures.[4]

Finally, the teams must present a section concerned with their proposed methodologies for collecting the data. These proposals are returned along with suggestions which can help the students avoid serious errors in their research design. We might add that having (fewer) teams rather than many individuals with whom to work enables the instructor to give the teams more feedback and individual assistance. More important, the team approach enables the students to learn from each other, motivate each other, and also to divide some of the work among themselves.

### *Questionnaire Construction and Administration*

Next, each team constructs a questionnaire designed to operationalize each variable in their hypotheses as well as the "test" factors they wish to include in their study. These questionnaires are pretested by fellow students and submitted to the instructor for any additional clarification that may be needed. Importantly, the students learn during this process to "precode" their question items for subsequent computer usage. At last, the students are ready to actually collect data. . . data in which they have become quite involved, interested, and curious.

At this point we should mention a compromise in sampling techniques that is necessitated by the time frame in which we must work. Collecting data from even a random sample of students or classes of students is a major undertaking, to say nothing of coordinating the efforts of eight or so teams, getting permission to survey classes, etc. For this reason we only survey our own research classes (and occasionally an additional sociology class). While this is not the ideal in terms of representativeness or in terms of having a larger number of cases, it *is* practical and has some advantages. Besides efficacy, students see how their fellow students have formulated their questionnaires. This leads to interesting discussions concerning the rationale and justification concerning many question items.

This approach also allows us to develop a *single* questionnaire, which includes *all* the demographic question items needed by the various teams, rather than include such variables in each separate questionnaire. In fact, the students learn to write and read coding instructions during this process. Each student then key punches his own demographic questionnaire and duplicates this IBM card for each student in the class. It is important for validity purposes that the students learn to administer the questionnaires, including the demographic questionnaires, in such a manner that the students know their responses are completely anonymous. Eventually, the demographic cards are matched with each student's questionnaire via a code number known only to the individual student.

Through these procedures the students have data from their well-constructed questionnaires in hand by the end of the first quarter. They also learn to code and keypunch data so that the second quarter can be devoted to learning the skills necessary to use the computer as a tool in learning to analyze their data.

### Second Quarter: "Data Analysis"

At the onset of the second quarter, the students learn to keypunch the raw data from their questionnnaires. They must also learn the deck set-up formats as a starting point for subsequent computer programs.[5] Learning computer skills can indeed occupy a full quarter in-and-of itself. Two major breakthroughs have allowed students to learn the necessary deck set-up and data modification procedures, as well as commonly used statistical subprograms. First, efficient and relatively easy to learn and use statistical packages have been developed for the social sciences (e.g., Nie, et al., 1975). Numerous pre-programmed subroutines provide a great deal of standardization necessary in order to teach and learn computer skills efficiently; yet, the necessary flexibility for analysis of the data is easily incorporated into the program design. As noted previously these advances along with concurrent advances in computer technology have given sociology a tremendous thrust forward.

#### *Computer Printout Guide*

Secondly, we have developed for this second course a format that enables the student to learn to use these skills much more readily. A condensed version of a recent "Student Study" of students' needs and values (Brown and McQuilkin, 1975) is used to illustrate the step-by-step process the students can follow, to a large degree, by simply plugging their own data into these illustrated steps.

In actuality, each student is given a computer printout which illustrates each step of the deck set-up of the data necessary for the student to get his data on the computer. These printouts illustrate necessary job control language cards, a variable list format, an input format for reading the raw data, variable and value labeling procedures, handling missing data procedures, various data modification techniques, index construction formats, and other optional tasks. In addition, "comment" cards which explain each step are included in the computer printouts.[5]

Of course these printouts are discussed in detail in class. Since the students are quite familiar with the "Student Study" questionnaire from first quarter discussions, they can more meaningfully follow their printout copies. Since each student has a printout before him or her, he or she can follow the discussion of each step, making additional marginal notes as needed. We have found that this "wholistic" or "A-to-Z," step-by-step approach greatly facilitates the students' learning to use the computer. Furthermore, it requires but minimal transfer by the student to insert his own data variables into this step-by-step guide. This latter process is, of course, what really reinforces the student for he or she is then actually *using* the computer on a set of his or her *own data* of interest. Again, team members teach each other and are able to reduce their work somewhat by a division of labor.

#### *Learning Data Analysis Via the Computer*

Perhaps the success of this approach is most obvious when an excited team of students proudly displays its first successful computer printout. Having (fewer) teams with which to work rather than numerous individuals enables the instructor to troubleshoot as the teams encounter errors in the process of learning these skills. It is important that the students receive sufficient assistance at this stage for even a simple error can create a great deal of frustration if the still novice students must find all their own errors.

Once the students have their own programs running and "cleaned," it is relatively simple for them to add new subprograms to their repertoire. Again, the example printout is a tremendous aid for the students not only see and study the actual program necessary to produce each procedure, but they can study brief examples of the computer output from each of these procedures. Since they have become familiar with the "Student Study" data, the output takes on much more meaning for them. Learning to analyze social data is quite difficult; therefore, having class discussions about what these output data are "saying" is most helpful. This approach gives the student examples and analytic principles he or she can apply to his or her own output.

### *Principal Statistical Procedures*

The student's first computer subprogram is, of course, to output frequency distributions for all the variables in his questionnaire. Once obtained, the student learns to "read" his data with respect to understanding the appropriate measures of central tendency and dispersion as well as to interpret the frequency and percentage data. Generally the student must learn the rationale and techniques necessary to collapse some of his variables into fewer categories for crosstabular analysis. He also learns some basic ways to construct social indices for data reduction purposes.

Next the student learns to crosstabulate his or her data and, importantly, to "read" correctly the tables that are output. In addition, students learn to select appropriate measures of association for interpreting the relationships they are analyzing. As the student learns to use other subprograms (e.g., Pearsons correlations, non-parametric associative statistics, scattergrams), he or she is shown the correspondence between these various modes of data analysis.

Other subprogrammed statistical procedures the students learn involve a mean breakdown of dependent variables by the independent variables and tests for linearity for the relationships at hand. They also learn partial correlation subprograms, tests of significance such as t-tests and oneway analysis of variance. Importantly they learn what these subprograms can tell them about their study and, more specifically, about their hypotheses.

Other important aspects of using this computer approach in teaching social research methods include, in addition to learning valuable computer skills:

1. The approach permits a means by which more highly motivated students can go as far as they wish with additional computer subprograms and analyses. The text and the example printouts, together with some assistance from the instructor, allow motivated students to learn beyond what can be expected of the average student (e.g., basic modes of multiple regression analysis, two-way analysis of variance, factor analysis, etc.).
2. A great deal of flexibility is possible in terms of establishing criteria for grading the students' work. For example, one could apply a "contract" system allowing a student to earn the grade he or she desires by successfully running the computer subprograms and analysis presentation for which he or she "contracts."
3. Most importantly, the students gain a feeling of accomplishment as well as a sense of competence by learning "advanced" techniques in social research while still an undergraduate. Invariably, ten to twenty percent of the undergraduates follow these two courses with independent studies and/or become research assistants within the department. A gratifying by-product of this computerized teaching approach has been the high degree of success our research oriented students have had in obtaining hard-to-come-by financial assistance in major university graduate programs.

In sum, we have found that undergraduate students, given the short-cut learning aids described above, *can* learn the computer skills mentioned, and that using these skills is instrumental in learning how to effectively analyze and interpret data. Most students are challenged by this approach. The less ambitious students can "contract" for less stringent goals. These students, however, pick up enough information to enable them to understand social data beyond what they could grasp by the simple textbook-lecture approach.

### Data Presentation

In that this subject is in itself complex (and usually under-taught), it should be noted that students are expected to systematically present the findings from their computer analyses in a reasonably sophisticated report. Their earlier proposal, of course, serves as a starting point. Considerable assistance is given to the teams as they learn to systematically pull their findings together, for this too is a difficult skill to learn for the novice writer. The "Student Study" final report, on reserve in the library, offers an example with which the students can relate.

### Summary

In brief, this paper has described a systematic approach for teaching undergraduate sociology students social research through student use of packaged computer programs while they carry out their own research projects. A number of schemes were described that enable this ambitious approach to lie within the bounds of practicality. As stressed early in the paper, the need for far greater emphasis on social research is growing. While it is not suggested that sociology departments mandate their undergraduate students to take more than three social research related courses, it is strongly suggested that, with the possible exception of small departments, additional social research electives be offered on a regular basis.

As a starting point, the data collection-computer data analysis approach as overviewed herein offers one workable alternative toward meeting the challenges that lie just ahead in sociology. Given our present level of computer technology, cautiously adding a touch of methodological innovation and training, we can indeed expect a surge in the rate and quality of functional information which sociologists can provide for those in positions of authority.

### Notes

1. Correspondence with regard to the contents of this paper should be directed to Professor Cook.

2. While the data are based on FTU curricula, these proportions appear to be fairly typical of other universities. We make no attempt here to condemn or defend the reasons for these notable differences in research emphasis.

3. This finding was noted in an editorial article, "Undergraduate Sociology Curriculum Emphasizes 'Practical' Course," in *ASA Footnotes*, Vol. 3, No. 4 (April), 1975. The article was excerpted from a study by CRM Books. As might be expected, very few (2 percent) of the two-year colleges offer social research courses.

4. In addition to lectures and discussions on each area, the students have regular reading assignments in a methods text. *Strategies of Social Research: The Methodological Imagination*, by H. W. Smith, Prentice-Hall, 1975, is presently being used for the first quarter course.

5. The text for this course is *SPSS: Statistical Package for the Social Sciences*, Second Edition, by Norman Nie, et al., McGraw-Hill, 1975.

6. Copies of these printouts and the questionnaire are available from the authors.

### References

Babbie, Earl R. (1975) *The Practice of Social Research.* Belmont, Calif.: Wadsworth.

Blalock, H. M., Jr. (1975) "Blalock Replies to Duncan on Methodology." *American Sociological Association Footnotes*, Vol. 3, No. 4 (April) 1975.

Brier, Alan and Ian Robinson (1973). *Computers and the Social Sciences.*

Brown, William R. and Paul McQuilkin. (1975) *FTU Students' Needs, Values, and Motivations in Academic and Non-Academic Areas.* Monograph, Florida Technological University Library, Orlando, Florida.

Nie, Norman H., C. Hadlai Hull, Jean G. Jenkins, Karen Steinbrenner, and Dale H. Bent. (1975) *Statistical Package for the Social Sciences*, Second Edition. New York: McGraw-Hill.

Smith, H. W. (1975) *Strategies of Social Research: The Methodological Imagination.* Englewood Cliffs, N.J.: Prentice-Hall.

# Integrating Prepackaged Computer Programs Into an Undergraduate Introductory Statistics Course

**Frank Scalzo**[1]
*Department of Mathematics*
*Queensborough Community College*
*Bayside, New York 11364*
*(212) 631-6364 or 423-3539*

**Rowland Hughes**
*Fordham University*
*113 West 60th Street*
*New York, New York 10023*

## Introduction

Skavaril (1974) of Ohio State University has noted the lack of the development of CAI instructional programs in elementary statistics:

> Over twenty CAI programs in statistics have been reported (Lekan, 1971); however, the majority of these programs are short course segments on a variety of specific topics and written in various languages for various computers.
>
> To the best of my knowledge, there have been no reports of a combined use of time-sharing computers to establish a complete computer-based system for the instruction of introductory statistics, a base which combines the technology of CAI, computer generation of exercises and answers, and use of the computer by students for analyses [p. 33].

A subsequent (1976) review of the literature by the authors did reveal, however, some integration of prepackaged statistical programs into full statistics courses on the elementary or intermediate level.

Kowalski (1972) reported the implementation of the computer in a statistics course for graduate students in the School of Dentistry at the University of Michigan. One of the objectives of this course—design and analysis in research—was to employ the use of computers for descriptive statistical procedures on numerical data and use this information to draw statistical inferences.

Tanis (1973) modified a standard two-semester course in mathematical probability and statistics at Hope College in Michigan by adding a two-hour computer laboratory to each semester of the course. In this computer laboratory, students utilized an IBM 1130 computer and a number of prepackaged computer programs to make empirical applications to many of the theoretical concepts which were proved in class.

During the 1973 Conference on Computers in the Undergraduate Curricula held at Claremont, California, Tanis noted:

> A valuable by-product of the statistics laboratory is the opportunity for students to do some research as undergraduates. Their mathematical maturity and background may not be sufficient to prove certain results theoretically but they are able to obtain answers empirically. . . A computer laboratory in statistics provides new and interesting challenges to both student and teacher. A lot of extra time is required. However, if the exercises are meaningful and not just busywork, the time is well spent [p. 10].

Ozarowski (1973) designed a CAI statistics curriculum using the Coursewriter interactive language to teach statistics via a remote teletype terminal and the APL programming language to enter and manipulate statistical data. After designing the CAI statistics curriculum he used the principle of matched pairs to randomly select the experimental and control groups. Ozarowski implemented his CAI statistics curriculum with the experimental group. Only minimal classroom instruction concerning the inputting of data was presented to the experimental group, who received instruction from a teaching program written in the Coursewriter language and ran sample problems using prepackaged statistical programs written in the APL language. The emphasis of the Coursewriter teaching program was on the understanding, interpretation, and application of statistical concepts.

Ozarowski ascertained that learning did take place in the experimental group; learning in the experimental group was greater than that in the control group; and the experimental mean course completion time was less than that of the control group. The practice of computation by use of statistical formulas was not required for the experimental group; therefore he concluded that the ability to compute by use of statistical formulas was not a requirement for learning and applying statistical concepts.

The Committee on the Undergraduate Program in Mathematics (CUPM, 1975) stressed the use of computers as an integral part of the undergraduate elementary statistics curriculum:

> A computer can serve three broad roles in the implementation of the goals of an introductory statistical course. It can: (1) clarify certain key ideas of the course, (2) perform routine numerical calculations, and (3) facilitate more active student participation in the development of statistical concepts [p. 509].

For several years the authors experimented with one aspect of the dialogue mode of CAI, where students interact with the computer via the use of prepackaged computer programs. This approach was initially used by Scalzo to present certain topics in several mathematics courses—elementary statistics, calculus, and technical mathematics—at Queensborough Community College of the City University of New York.

Some faculty members were of the opinion that there are several advantages to this prepackaged program approach: (1) The prepackaged programs conserve the instructor's time spent on preparing problems; (2) the students are, in general, able to cover the same number of topics as in a traditionally taught course while learning about the use of computers; (3) students are afforded an opportunity to solve a greater variety of more complex problems; and (4) the prepackaged programs provide the instructor with correct final calculations for the assigned problems.

Prior to developing the full course described later in this paper, Scalzo had also been using an algorithmic approach to the instruction of mathematics in which flowcharting and prepackaged computer programs are used as tools in solving mathematical problems. Flowcharts or diagrams of the step-by-step procedure used to solve a specific type of mathematical problem are developed in class. After the students have shown that they are able to carry out the algorithm manually, they are directed to use prepackaged computer programs to solve similar problems.

Subsequent discussions were held with the mathematics professor at Queensborough in whose classes certain units on statistics had been presented by Scalzo with the CAI approach. Encouraged by them and some favorable feedback of students' reactions, Scalzo began to formulate and teach a one-semester undergraduate course, Mathematics 36: Elementary Computer-Assisted Statistics. The prerequisite for this course is completion of Mathematics 20: Intermediate Algebra or a satisfactory score on the mathematics department's *Omnibus Placement Test*. Although this course satisfies Queensborough's general liberal arts requirement in mathematics for all students, a large majority who enroll in the course are pursuing a two-year program in environmental health or one for medical laboratory technician.

## Description of Mathematics 36

After discussing the possible course contents of Mathematics 36, the authors finalized a curriculum for the course which includes introduction to flowcharting, the BASIC programming language, manual calculation of statistical problems, and fourteen prepackaged computer statistical programs with thirty-nine applicable problems sets. This course has been offered at Queensborough for the past five semesters to one secion of approximately twenty-five students. Increased enrollment due to student interest has resulted in plans to offer two sections of Mathematics 36 in the future.

The course is comprised of the following seven topic units: Understanding the Use of Computers; Descriptive Statistics; Sets, Permutations, and the Binomial Theorem; Elementary Probability Concepts; Random Variables and Normal Distributions; Hypothesis Testing; and Additional Statistical Techniques (Chi-square, Linear Regression, Coefficient of Correlation, and Analysis of Variance).

### *Prepackaged Programs*

The following fourteen prepackaged programs are integrated in the topic units mentioned above.

STAT1: Ungrouped Data
(This program computes the arithmetic mean, variance, and standard deviation of N numbers.)

STAT2: Grouped Data
(This program computes the mean, median, variance, and standard deviation for a set of grouped data.)

STAT3: Counting Program
(This program computes the number of permutations of r objects selected from n distinct objects or the number of combinations of r objects selected from n distinct objects.)

STAT4: The Binomial experiment
(This program computes the probability of r successes in n independent trials of a binomial experiment.)

STAT5: Computing the Z-Score/Scores When Using the Normal Curve to Approximate Binomial Probabilities
(This program computes the Z-score/scores and tells which area to compute when using the normal curve to approximate binomial probabilities.)

STAT6: Normalizing Raw Test Scores
(This program normalizes a set of n raw scores where $n \leq 30$.)

STAT7: Computing Confidence Intervals for the Mean of a Population
(This program computes confidence intervals for the mean of a population.)

STAT8: Computing Observed Z-score of Observed t-score for Differences Between a Sample Mean and a Population Mean
(This program computes observed Z-score or observed t-score for differences between a sample mean and a population mean.)

STAT9: Computing the Observed Z-score or t-score for Differences Between Two Sample Means
(This program computes observed Z-score or observed t-score for differences between two sample means.)

STAT10: Computing the Observed Z-score and 95% Confidence Interval in Testing Proportions for a Binomial Experiment
(This program computes the observed Z-score and 95% confidence interval for tests of proportionsin a binomial experiment.)

STAT11: Computing the Observed Chi-Square Value
(This program computes the observed Chi-square value for a given set of observed frequencies.)

STAT12: Finding the Estimated Linear Regression Equation
(This program computes the slope and y-intercept for an estimated linear equation.)

STAT13: Computing the Coefficient of Correlation
(This program computes the coefficient of correlation for a set of ordered pairs [x,y], where x is the independent variable and y is the dependent [predicted] variable.)

STAT14: Computing the Observed F (variance) Ratio
(This program computes the calculated F ratio in an analysis of variance procedure.)

The materials for each of the fourteen prepackaged statistical programs include an instruction sheet, a flowchart, coding in BASIC, and a problem set. Scalzo and Hughes (1976) have formulated these prepackaged statistical programs and described their use as a component of a one-semester computer-assisted course in a recently published text *Elementary Computer-Assisted Statistics*.

To illustrate the use of the programs in Mathematics 36, two examples are given in the next section. The first example shows how the first prepackaged program in the course is used in conjunction with learning some basic topics of descriptive statistics—the arithmetic mean, variance, and standard deviation of N numbers—for a set of ungrouped data. The second example describes how a combination of three prepackaged programs is used to teach hypothesis testing. One program—STAT9—is shown in detail for computing the observed Z-score or the observed t-score for differences between two sample means.

## Examples

INSTRUCTIONS FOR STAT1

NAME: STAT1

DESCRIPTION: This program computes the arithmetic mean, variance and standard deviation of N numbers.

INSTRUCTIONS: Enter the data beginning in line 150 as follows:

150 DATA N, $X_1$, $X_2$, $X_3$, ...., $X_n$

where: N represents the number of raw scores.

$X_1$, $X_2$, ...., $X_n$ are the raw scores.

Note: Line 200 is the END statement.

EXAMPLE:

User's Input

```
GET-STAT1
150 DATA 5, 70, 90, 100, 40, 50
151 DATA 5, 70, 68, 72, 67, 73
```

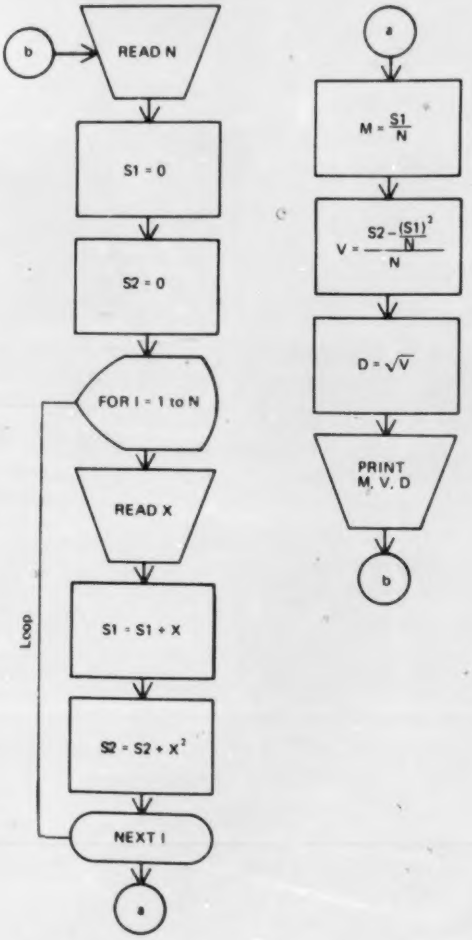
Computer's Output

```
STAT1
MEAN = 70  VARIANCE = 520 ST. DEV. =
22.8035
MEAN = 70 VARIANCE = 5.2 ST. DEV. =
2.28035
OUT OF DATA IN LINE 5
```

CODING OF STAT1

```
 5 READ N
10 LET S1=0
15 LET S2=0
20 FOR I=1 TO N
25        READ X
30        LET S1=S1+X
35        LET S2=S2+X↑2
40 NEXT I
45 LET M=S1/N
50 LET V=(S2-(S1)↑2/N)/N
55 LET D=SQR(V)
60 PRINT   "MEAN="M;   "VARIANCE="V;   "ST.
DEV.="D
65 GO TO 5
150 DATA
200 END
```

### FLOWCHART FOR STAT1



### Problem Set—STAT1

1. The following final examination scores for two different sections of statistics are listed below:

| *Section 1* | *Section 2* |
|---|---|
| 75 | 75 |
| 70 | 55 |
| 82 | 95 |
| 81 | 80 |
| 68 | 81 |
| 70 | 82 |
| 70 | 70 |
| 75 | 91 |
| 81 | 67 |
| 82 | 52 |

Use STAT1 to compute $\bar{X}$ (mean), $s^2$ (variance), and s (standard deviation) for each section.

2. Use STAT1 to compute $\bar{X}$, $s^2$, s for the five daily stock prices ($31.85, $31, $31, $31.15, $31.50).

3. Use STAT1 to compute $\bar{X}$, $s^2$ and s for the following sampling of twenty senior grade point averages at Orange University: 3.2, 2.5, 3.1, 2.1, 3.3, 3.7, 1.9, 2.2, 2.2, 2.6, 2.1, 2.9, 2.8, 3.6, 3.0, 2.0, 3.5, 2.2, 2.6, 3.1.

4. Use STAT1 to compute $\bar{X}$, $s^2$, and s for the following amounts which represent a truck driver's weekly gross earning for the past eight weeks: $257, $315, $291, $201, $315, $288, $352, $299.

As can be seen from the list of the prepackaged programs given earlier, the students are then introduced to a specific sequence of topics which generally deals with various basic statistical concepts. Once the students have mastered the computation of the observed Z-score or observed t-score for differences between a sample mean and a population mean, they proceed to a unit on hypothesis testing. They develop in class a general algorithm used in conjunction with STAT8, STAT9, or STAT10. This algorithmic approach provides students with a simple, step-by-step procedure they can easily follow for hypothesis testing.

Let us look at this algorithm and the prepackaged program materials for STAT 9, which is used to test hypotheses concerning two sample means.

### *An Algorithm for Hypothesis Testing*

1. State the motivated hypothesis ($H_m$) and the null hypothesis ($H_o$).
2. Decide whether you are using a one-tail test or a two-tail test.
3. Decide whether to use a critical Z-score ($N \geq 30$) or a critical t-score ($N<30$).
4. Find the critical Z-score (scores) or the critical t-score (scores) by looking at the appropriate table.
5. Compute the observed Z-score or the observed t-score. Use STAT8, STAT9, or STAT10 to compute the observed Z-score or the observed t-score.
6. State your conclusion (i.e., we can reject $H_o$ or we cannot reject $H_o$).

INSTRUCTIONS FOR STAT9

NAME: STAT9

DESCRIPTION: This program computes observed Z-score or observed t-score for differences between two sample means.

INSTRUCTIONS: Enter the data beginning in line 150 as follows:

```
150 DATA M1, M2, V1, V2, N1, N2
```

where:

M1 is the first sample's mean;
M2 is the second sample's mean;
V1 is the first sample's variance;
V2 is the second sample's variance;
N1 is the size of the first sample;
N2 is the size of the second sample.

Note: Line 200 is the END statement

EXAMPLE:

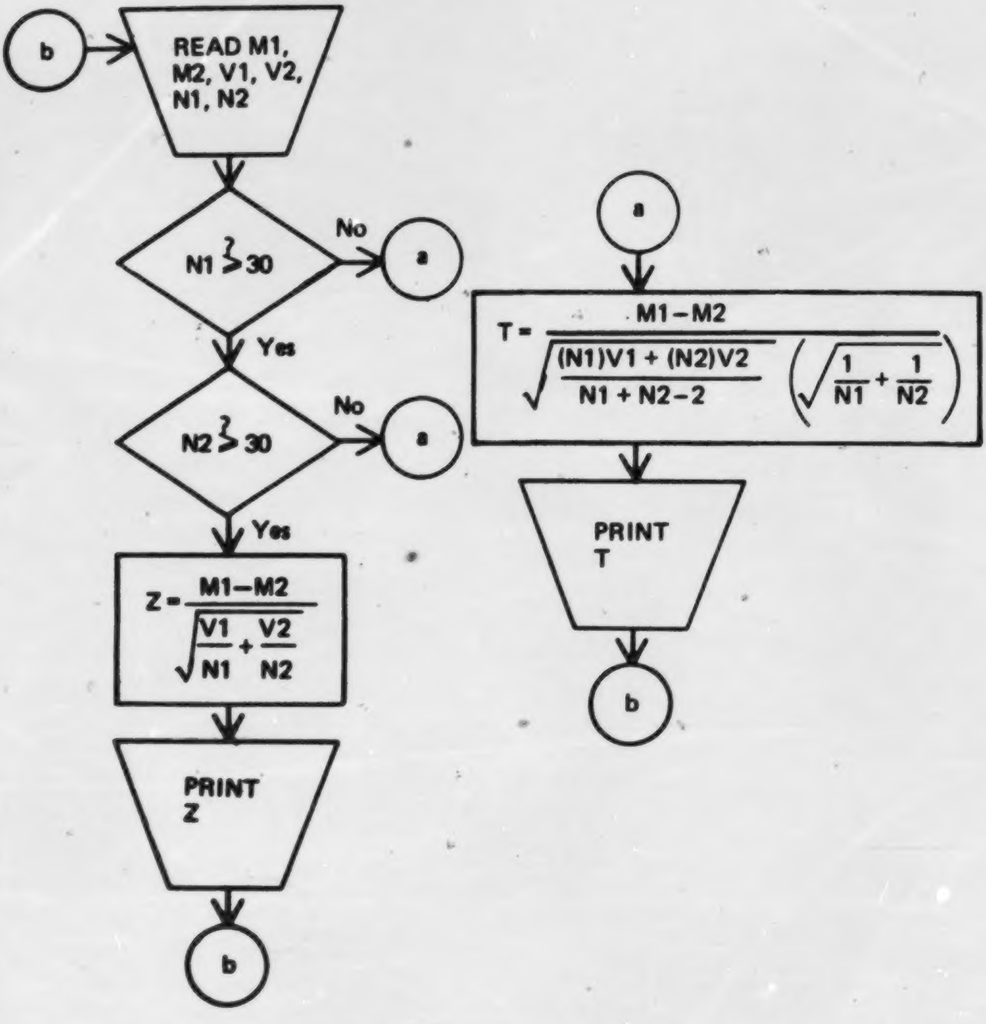
User's Input

```
GET-STAT9
150 DATA 108.7, 110.4, 62.5681, 77.44, 174, 150
151 DATA 74.4, 71.8, 262.64, 160.56, 5, 5
RUN
```

Computer's Output

```
STAT9
OBSERVED Z= -1.81649
OBSERVED T = 0.252774
OUT OF DATA IN LINE 5
```

# FLOWCHART FOR STAT9


b
READ M1, M2, V1, V2, N1, N2
N1 ≥? 30
No
a
Yes
N2 ≥? 30
No
a
Yes
$Z = \frac{M1-M2}{\sqrt{\frac{V1}{N1}+\frac{V2}{N2}}}$
PRINT Z
b
a
$T = \frac{M1-M2}{\sqrt{\frac{(N1)V1+(N2)V2}{N1+N2-2}}\left(\sqrt{\frac{1}{N1}+\frac{1}{N2}}\right)}$
PRINT T
b

CODING OF STAT9

```
  5 READ M1. M2. V1. V2. N1. N2.
 10 IF N1>=30 THEN 30
 14 LET D=SQR(((N1)*V1+(N2)*V2)/
    N1+N2-2))*SQR(1/N1+1/N2)
 15 LET T=(M1-M2)/D
 20 PRINT “OBSERVED T=“T
 25 GO TO 5
 30 IF N2>=30 THEN 40
 35 GO TO 14
 40 LET Z=(MI-M2)/SQR(V1/N1+V2/N2)
 45 PRINT “OBSERVED Z=“Z
 50 GO TO 5
150 DATA
200 END
```

## Problem Set—STAT9

Use the algorithm for hypothesis testing and STAT9 as tools to solve the following problems:

1. The City of Hoboken, New Jersey has two junior high schools, Demarest Junior

High and Brandt Junior High. A random sampling of six ninth grade student IQ scores are selected from each of the two junior high schools. The results are as follows:

| *Demarest IQ Scores* | *Brandt IQ Scores* |
|---|---|
| 100 | 100 |
| 107 | 105 |
| 110 | 111 |
| 95 | 90 |
| 90 | 92 |
| 120 | 115 |

You assume that the sampling above indicates that, on the average, Demarest Junior High School's ninth grade students have higher IQ scores than Brandt Junior High's ninth grade students. Test your hypothesis at the .05 level of significance.

2. Two machines A and B produce 15 cent candy bars for the Cavity Candy Company. A random sampling of 100 candy bars produced by machine A shows a mean weight of 3.2 ounces with a variance of 0.0289 ounces. A random sampling of 90 candy bars produced by Machine B determines a mean weight of 3.15 ounces with a variance of 0.0225 ounces. You suppose that the samples above indicate that, on the average, machine B produces candy bars of less weight than those produced by Machine A. Test your hypothesis at the .05 level of significance.

3. Test the hypothesis in exercise 2 above at the .01 level of significance.

4. Given the following information for two random samples:

SAMPLE 1: $\bar{Y}_1 = 50$
SAMPLE 2: $\bar{Y}_2 = 46{:}174$

$S_1{}^2\,2 = 100$
$S_2{}^2\,2 = 95$

$n_1 = 35$
$n_2 = 40$

(a) Test the motivated hypothesis $H_m$: $\bar{Y}_1 \neq \bar{Y}_2$ at the .01 level of significance.

(b) Test the motivated hypothesis $H_m$: $\bar{Y}_1 \neq \bar{Y}_2$ at the .05 level of significance.

5. Machines A, B, and C manufacture the same type of pipe for the Wet Plumbing Company. The following table indicates a random sampling of pipes and their diameters made by machines A, B, and C:

| *Machine* | *Number in Sample* | *Mean Diam. in Inches* | *Variance of Diameter* |
|---|---|---|---|
| A | 35 | 2.03 in. | 0.001 |
| B | 40 | 2.032 in. | 0.015 |
| C | 35 | 2.038 in. | 0.030 |

Test the following motivated hypotheses at the .05 level of signifcance:

(a) $H : \bar{A} < \bar{B}$
(c) $H : \bar{A} \neq \bar{B}$
(b) $H : \bar{C} > \bar{B}$
(d) $H : \bar{A} = \bar{C}$

Now repeat each of the tests above at the .01 level of significance.

6. Given the following information for two random samples:

SAMPLE 1: $\bar{Y}_1 = 75$ $S_1{}^2\,2 = 100$ $n_1 = 10$
SAMPLE 2: $\bar{Y}_2 = 80$ $S_2{}^2\,2 = 68$ $n_2 = 8$

Test the motivated hypothesis $H_m$ : $\bar{Y}_1 < \bar{Y}_2$ at the .01 level of significance.

## Course Evaluation

During the spring 1976 semester, Scalzo as instructor implemented this computer-assisted statistics curriculum with one class of twenty-six liberal arts students at Queensborough Community College of The City University of New York. He administered three unit tests and a final examination to the students enrolled in this CAI course. The questions on the unit tests and the final examination were selected from *Johnson's Test Item Card File for Elementary Statistics* (1974).

An *Observation Rating Scale for Psychomotor and Conceptual Learning Difficulties* was formulated by the authors and subsequently validated by three experts in the field of testing and computer science. This validated instrument was used to determine conceptual and psychomotor learning difficulties encountered by students when operating computer terminals.

Hughes interviewed Scalzo and each of the students who completed the course outside the class situation, he asked each student twenty questions concerning their perceptions of the course presentation, content, and sequence.

Some of the major conclusions of this course evaluation were:

1. Results of the scores on the unit tests and the final examination indicated that a large majority of the students who completed the computer-assisted statistics course succeeded on each of the unit tests and the final examination.

2. Learning in this class of CAI students, appeared to become more effective, in general, as the course progressed.

3. Although most of the students in this CAI course initially had some psychomotor and conceptual learning difficulties while operating a computer terminal, they appeared to overcome reasonably these difficulties after two or three practice sessions.

4. A large majority of the students were of the opinion that flowcharting and prepackaged computer programs were useful as practical tools in the solution of statistical problems.

5. A small majority of the students preferred to take a CAI statistics course rather than a traditional statistics course.

6. The instructor and the students who completed this CAI course appeared to be in general agreement that the course was presented in a logical and sequential order.

In light of the findings of this course evaluation, the authors recommend that this CAI statistics curriculum be implemented and evaluated using sample student populations selected from four-year institutions as well as other two-year colleges.

## Note

Correspondence with regard to the contents of this paper should be directed to Professor Scalzo.

## References

Committee on the Undergraduate Program in Mathematics. *A Compendium of CUPM Recommendations.* Washington, D.C.: Mathematical Association of America, 1975.

Johnson, Robert. *Test Item Card File for Elementary Statistics.* North Scituate, Mass.: Ducksbury Press, 1974.

Kowalski, Charles J. "Computer Aided Instruction of Statistics to Dental Students." *Journal of Dental Education,* 36:68-73, January 1972.

Lekan, H. (ed.). *Index to Computer Assisted Instruction.* New York: Harcourt Brace Jovanovich, Inc., 1971.

Ozarowski, Peter C., Jr. "A Study in the Design and Implementation of a Course in the Basic Fundamentals of Statistics Via a Computer." Unpublished doctoral dissertation, University of Alabama, Montgomery, Alabama, 1973.

Scalzo, Frank and Rowland Hughes. *Elementary Computer-Assisted Statistics.* New York: Mason and Charter, Inc., Petrocelli Books, 1976.

Skavaril, Russell V. "Computer-Based Instruction of Introductory Statistics." *Journal of Compter-Based Instruction,* 1:32-39, August 1974.

Tanis, Elliot A. *A Computer Laboratory for Mathematical Probability and Statistics.* Washington, D.C.: National Science Foundation, 1973.

# A Computer-Based Laboratory for Mathematical Statistics and Probability

**Elliot A. Tanis**
*Hope College*
*Holland, Michigan 49423*
*(616) 392-5111 Ext. 2267*

## Introduction

Hope College is a four-year, undergraduate, liberal arts college of 2,100 students. Most of the students who major in mathematics or computer science and some students who major in one of the natural sciences, psychology, or economics take a year-long, calculus-based course in mathematical statistics and probability.

During the last six years we have developed an optional computer-based laboratory that a student may elect to take in conjunction with the statistics course. Earlier papers [2,3] described the development of the laboratory and gave an outline of the topics that were covered. At that time Hope College owned an IBM 1130 computer. We now own a Xerox Sigma VI computer which has timesharing capabilities.

In this paper a summary of the earlier papers is given. In addition, recent developments and the availability of the materials is discussed.

## Background and Description of Laboratory

We began to develop the laboratory materials and supporting computer software during the summer of 1971. It was motivated by students who had taken our statistics course and were interested in using the computer for research projects. The projects often required some sort of simulation. In the process of writing a computer program to properly perform a simulation, the student not only had to understand the underlying theory, he also gained a greater appreciation for this theory. Webelieved that similar types of experiences for more of our students would be beneficial. This prompted the development of the laboratory.

Originally the laboratory was mandatory. It is now optional because (1) some students had scheduling conflicts, and (2) some students did not like a computer-based laboratory for various reasons.

The laboratory meets for a two-hour block each week for an extra hour of credit. The first hour is spent discussing the previous week's assignment and introducing the new assignment. During the second hour the students begin to write their programs.

It is now possible for students to run their programs through batch processing or interactively at a terminal.

## Laboratory Manual and Subprograms

An instructional package has been developed for a year-long laboratory for a mathematical statistics and probability course. The package includes a laboratory manual and supporting computer software.

The laboratory manual has gone through three revisions. It now contains more than 250 problems with complete solutions for twenty-five of them. The manual parallels the textbook *Probability and Statistical Inference* [1]. It can also be used with most of the standard textbooks.

The students write their own programs for solving the exercises. However over fifty subprograms are provided for their use. Among these are:

1. A pseudo-random number generator.
2. Graphing routines for: (1) the empirical distribution function with the option of

superimposing the theoretical distribution function, (2) a relative frequency ogive curve with the option of superimposing the theoretical distribution function, (3) a relative frequency histogram with the option of superimposing the theoretical probability density function, (4) a scatter diagram with the option of adding the least squares regression line, (5) a relative frequency power function with the option of superimposing the theoretical power function, and (6) printing the graph of any nonnegative valued function.

3. Programs that give the values of seven distribution functions.

4. Programs that give the values of the inverses of four distribution functions.

5. Programs that select random samples from the standard distributions.

It is possible to use all of these programs interactively at a terminal. The graphing routines use essentially a 100 x 50 character grid when run through batch processing and a 60 x 30 character grid when run interactively.

### Goals

Some of the goals of the laboratory are to:

1. Help the student understand and gain a greater appreciation of the theory.
2. Illustrate that certain probability models are appropriate for particular random experiments.
3. Show how the computer can be used as an aid for analyzing data.
4. Expand possibilities for undergraduate research projects.

### Examples

In this section two problems with complete solutions are given. These two problems were run using the interactive versions of RFHPF and CONIN. The subroutine RFHPF provides a Relative Frequency Histogram with the Probability Density Function superimposed. The subroutine CONIN depicts thirty CONfidence INtervals.

#### *Problem 1*

Let $X_1, X_2, X_3$ be a random sample of size 3 from a uniform distribution U(0,10). Thus $f(x) = 1/10$ and $F(x) = x/10$ for $0 < x < 10$.

(a) Define $g_1(y)$, $g_2(y)$, and $g_3(y)$, the marginal p.d.f.'s of the 3 order statistics, $Y_1 < Y_2 < Y_3$.

(b) Generate 200 samples of size 3, say $x_1, x_2, x_3$, from the uniform distribution U(0,10). For each sample let $y_1 = \min(x_1, x_2, x_3)$, $y_2 =$ median of $(x_1, x_2, x_3)$, and $y_3 = \max(x_1, x_2, x_3)$.

(c) Use RFHPF to plot a histogram of the 200 observations of $Y_1$ along with $g_1(y)$, find the sample mean and sample variance of these observations.

(d) Use RFHPF to plot a histogram of the 200 observations of $Y_2$ along with $g_2(y)$. Find the sample mean and sample variance of these observations.

(e) Use RFHPF to plot a histogram of the 200 observations $Y_3$ along with $g_3(y)$. Find the sample mean and sample variance of these observations.

#### *Problem 2*

Let X be N(16.4,2.9).

(a) Generate 50 samples of size $n = 16$ from this distribution. For each sample, find the values of $Y_5$ and $Y_{12}$, the 5th and 12th order statistics. Store these values in two arrays, say Y5 and Y12, which have been dimensioned to 50. (Use 30 for on-line version.) Hint: You may use the subroutine ORDER.

(b) Verify that $(y_5, y_{12})$ is a 92.32% confidence interval for $\pi_{.5} = 16.4 = \mu$.

(c) Approximately 92.32% of the 50(30) confidence intervals from part a should contain the median. To illustrate this, use the subroutine CONIN(14.0,18.8,Y5,Y12,16.4) to depict the confidence intervals.

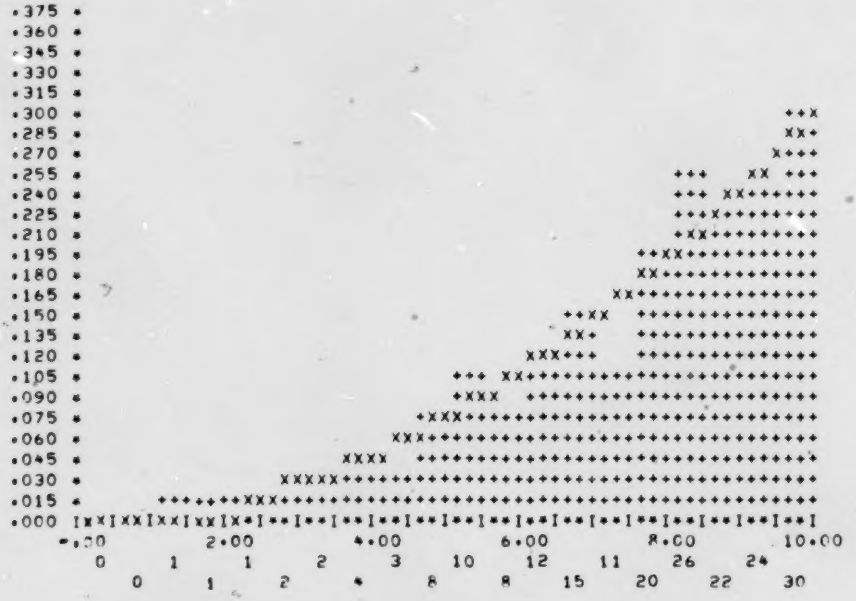
(d) What proportion of the confidence intervals contain the median?

```
C . . . SOLUTION FOR PROBLEM [redacted]1.
C
C . . . DIMENSION X TO HOLD EACH SAMPLE OF SIZE 3, Y TO HOLD THE
C . . . ORDERED SAMPLE OF SIZE 3, Y1,Y2,Y3 TO HOLD THE 200
C . . . OBSERVATIONS OF EACH ORDER STATISTIC.
C . . . DEFINE G1,G2,G3, THE P.D.F.'S OF THE THREE ORDER STATISTICS
C . . . Y1,Y2,Y3, RESPECTIVELY, AS STATEMENT FUNCTIONS.
C
      DIMENSION X(3), Y(3), Y1(200), Y2(200), Y3(200), NOBS(2C)
      G1(U) = 3.*((1. - U/10.)**2)*(1./10.)
      G2(U) = 6.*(U/10.)*(1. - U/10.)*(1./10.)
      G3(U) = 3.*((U/10.)**2)*(1./10.)
C
C . . . GENERATE 200 OBSERVATIONS OF Y1,Y2,Y3.  FIRST GENERATE THREE
C . . . OBSERVATIONS OF X.  USE THE SUBROUTINE ORDER TO OBTAIN THE
C . . . VALUES OF THE ORDER STATISTICS.
C
      DO 20 K=1,200
      DO 10 J=1,3
      X(J) = 10.*RAN(1)
   10 CONTINUE
      CALL ORDER(3,X,Y)
      Y1(K) = Y(1)
      Y2(K) = Y(2)
      Y3(K) = Y(3)
   20 CONTINUE
C
C . . . PLOT HISTOGRAMS OF THE OBSERVATIONS OF Y1, THEN Y2,
C . . . THEN Y3, WITH THE RESPECTIVE P.D.F.'S SUPERIMPOSED. ALSO
C . . . CALCULATE, USING SMASV, AND PRINT THE RESPECTIVE SAMPLE
C . . . MEANS AND SAMPLE VARIANCES.
C
      WRITE(108,850)
  850 FORMAT('1',//,8X,'OUTPUT FOR PROBLEM 5.5-1C',//)
      CALL RFHPF(1,20,0.0,10.0,0.375,200,Y1,NOBS,G1)
      WRITE(108,800)
  800 FORMAT('0',8X,'HISTOGRAM AND PDF FOR THE SMALLEST ORDER STATISTIC'
     *,/)
      CALL SMASV(Y1BAR,Y1VAR,200,Y1)
      WRITE(108,825) Y1BAR,Y1VAR
  825 FORMAT(' ',8X,'THE SAMPLE MEAN IS',F8.3,/,8X,'THE SAMPLE VARIANCE '
     *IS',F8.3)
      WRITE(108,860)
  860 FORMAT('1',//,8X,'OUTPUT FOR PROBLEM 5.5-1D',//)
      CALL RFHPF(1,20,0.0,10.0,0.375,200,Y2,NOBS,G2)
      WRITE(108,805)
  805 FORMAT('0',8X,'HISTOGRAM AND PDF FOR THE MEDIAN',/)
      CALL SMASV(Y2BAR,Y2VAR,200,Y2)
      WRITE(108,825) Y2BAR,Y2VAR
      WRITE(108,870)
  870 FORMAT('1',//,8X,'OUTPUT FOR PROBLEM 5.5-1E',//)
      CALL RFHPF(1,20,0.0,10.0,0.375,200,Y3,NOBS,G3)
      WRITE(108,810)
  810 FORMAT('0',8X,'HISTOGRAM AND PDF FOR THE LARGEST ORDER STATISTIC'
     *,/)
      CALL SMASV(Y3BAR,Y3VAR,200,Y3)
      WRITE(108,825) Y3BAR,Y3VAR
      STOP
      END
```

```
.375 *
.360 *                   OUTPUT FOR PROBLEM [redacted]-1C
.345 *
.330 *
.315 *++++++
.300 *++++++
.285 *XX++++
.270 *++X+++
.255 *+++XX+      +++
.240 *+++++XX     +++
.225 *++++++ X ++++++
.210 *++++++  XX+++++
.195 *++++++   +XX+++
.180 *++++++++++++XX+
.165 *++++++++++++++XX
.150 *++++++++++++++++XX
.135 *++++++++++++++++++XX
.120 *++++++++++++++++++  XXX
.105 *+++++++++++++++++++++  XX++
.090 *+++++++++++++++++++++   +XXX
.075 *++++++++++++++++++++++++++++XXX
.060 *++++++++++++++++++++++++++++++ XXX++
.045 *++++++++++++++++++++++++++++++++++XXXX++++
.030 *++++++++++++++++++++++++++++++++++++++XXXXX
.015 *+++++++++++++++++++++++++++++++++++++++++++XXX++
.000 I**I**I**I**I**I**I**I**I**I**I**I**I**I**I**I*XIXXIXXIXXIXXI
    -.00        2.00        4.00        6.00        8.00       10.00
      31    18    25    11    11     5     4     1     0     0
         31    23    15     7     7     6     4     1     0     0

      HISTOGRAM AND PDF FOR THE SMALLEST ORDER STATISTIC

      THE SAMPLE MEAN IS   2.320
     THE SAMPLE VARIANCE IS   3.297
```

```
.375 *                  OUTPUT FOR PROBLEM [redacted]-1D
.360 *
.345 *
.330 *
.315 *
.300 *
.285 *
.270 *
.255 *
.240 *
.225 *
.210 *
.195 *                                       +++
.180 *                                       +++
.165 *                        +++            +++
.150 *                       XXXXXXXXXXXXX   +++
.135 *                  XXXXX ++++++++++++XXXXX++++
.120 *               XXX+++   ++++++++++++   ++XXX+
.105 *            XXX++++++   ++++++++++++++++++++XXX    +++
.090 *      +++XXX+++++++++++++++++++++++++++++++++++XXX +++
.075 *      +XX+++++++++++++++++++++++++++++++++++++++++XX++
.060 *     XX+++++++++++++++++++++++++++++++++++++++++++++XX
.045 *   +X+++++++++++++++++++++++++++++++++++++++++++++++++X  +++
.030 *  XX++++++++++++++++++++++++++++++++++++++++++++++++++ XX+++
.015 * X ++++++++++++++++++++++++++++++++++++++++++++++++++++++X++
.000 IX*I**I**I**I**I**I**I**I**I**I**I**I**I**I**I**I**I**I**I*XI
    -.00        2.00        4.00        6.00        8.00       10.00
       0     9     9    12    16    15    10    14     8     2
          5     9    10     9    14    15    19     9    11     4

      HISTOGRAM AND PDF FOR THE MEDIAN

      THE SAMPLE MEAN IS   5.176
     THE SAMPLE VARIANCE IS   5.341
```

```
OUTPUT FOR PROBLEM 5[illegible]1E
```



```
HISTOGRAM AND PDF FOR THE LARGEST ORDER STATISTIC

THE SAMPLE MEAN IS    7.554
THE SAMPLE VARIANCE IS    3.497
```

```
C . . . SOLUTION FOR PROBLEM
C
C . . . DIMENSION X TO HOLD THE ORIGINAL SAMPLE AND Y TO HOLD
C . . . THE ORDERED SAMPLE OF SIZE 16.  DIMENSION Y5 AND Y12
C . . . TO HOLD THE 5TH AND 12TH ORDER STATISTICS FOR THE 30
C . . . (50 IF USING BATCH VERSION) SAMPLES OF SIZE 16 FROM
C . . . THE NORMAL DISTRIBUTION N(16.4,2.9).
C
      DIMENSION X(16), Y(16), Y5(30), Y12(30)
C
C . . . LET M=30 FOR TIME SHARING, 50 FOR BATCH.
C
      M = 30
      DO 10 K=1,M
C
C . . . USE EITHER THE SUBROUTINE NORM OR THE FUNCTION
C . . . SUBPROGRAM XNRML TO GENERATE THE SAMPLE OF SIZE 16
C . . . FROM THE NORMAL DISTRIBUTION N(16.4,2.9).
C
      CALL NORM(16.4,2.9,16,X)
C
C . . . USE THE SUBROUTINE ORDER TO ORDER THE RANDOM SAMPLE.
C
      CALL ORDER(16,X,Y)
C
C . . . FILL ARRAYS Y5 AND Y12 WITH THE 5TH AND 12TH ORDER
C . . . STATISTICS.
C
      Y5(K) = Y(5)
      Y12(K) = Y(12)
   10 CONTINUE
C
C . . . USE THE SUBROUTINE CONIN TO DEPICT THE CONFIDENCE
C . . . INTERVALS.
C
      WRITE(108,850)
  850 FORMAT('1',8X,'OUTPUT FOR PROBLEM 6.1-1',//)
      CALL CONIN(14.0,18.8,Y5,Y12,16.4)
      WRITE(108,800)
  800 FORMAT('0',5X,'EACH INTERVAL IS A 92.32% CONFIDENCE INTERVAL')
C
C . . . COUNT THE NUMBER OF INTERVALS THAT CONTAIN THE
C . . . MEDIAN (MEAN) 16.4.
C
      NUM = 0
      DO 20 K=1,M
      IF(Y5(K).LE.16.4.AND.16.4.LE.Y12(K)) NUM = NUM + 1
   20 CONTINUE
      PROP = FLOAT(NUM)/FLOAT(M)
      PERCENT = 100.*PROP
      WRITE(108,805) PERCENT
  805 FORMAT('0',5X,F5.2,' PER CENT OF THE INTERVALS CONTAIN THE MEDIAN
     *')
      STOP
      END
```

```
       OUTPUT FOR PROBLEM



             CONFIDENCE INTERVALS
             *****************************           (15.107,17.815)
                 *************                       (15.543,16.727)
             **************************              (15.117,17.567)
                           *********************     (16.452,18.414)
               *************************             (15.303,17.614)
                ********************                 (15.439,17.289)
              *************************              (15.200,17.597)
                  ***************                    (15.665,17.012)
               ********************                  (15.298,17.205)
              ***************************            (15.238,17.718)
           ************************                  (14.954,17.146)
                 **********************              (15.534,17.563)
             ***********************                 (15.176,17.291)
         ***************************                 (14.783,17.272)
             ************************                (15.171,17.343)
       *********************                         (14.579,16.479)
               ************************              (15.332,17.536)
                 ***********************             (15.580,17.606)
               ********************************      (15.389,18.282)
                     *************************       (15.917,18.221)
                 *************************           (15.552,17.820)
                   ************                      (15.695,16.809)
               **************************            (15.301,17.776)
           *******************************           (14.958,17.814)
                   *********************             (15.761,17.641)
                  ************                       (15.657,16.681)
                       ****************              (16.145,17.561)
                ********************                 (15.411,17.269)
              ***************                        (15.266,16.600)
                           ******                    (16.483,17.007)
*+---------+---------+----X----+---------+---------+
 14.00   14.96     15.92     16.88     17.84     18.80
PARAMETER=  16.400

     EACH INTERVAL IS A 92.32% CONFIDENCE INTERVAL

     93.33 PER CENT OF THE INTERVALS CONTAIN THE MEDIAN
```

## Conclusions and Recommendations

This laboratory provides a worthwhile experience for the student. The interactive capability is beneficial for class demonstration and it also allows a student to obtain immediate output. Some students using teletype terminals that print at ten cps have a tendency to tie up the terminal, especially when several graphs are required. Thus batch processing is still the best for some solutions and for some students.

Graphics terminals could have a positive impact on this laboratory. Unfortunately, Hope College does not yet own a graphics terminal, although proposals have been written which include the purchase of graphics terminals.

## Distributions of Materials

The laboratory manual and subprograms have been favorably reviewed by CONDUIT. Copies of the laboratory manual and a tape containing the subprograms are available from CONDUIT, P.O. Box 388, Iowa City, Iowa 52240.

## References

1. Robert V. Hogg and Elliot A. Tanis, *Probability and Statistical Inference*, Macmillan Publishing Co., Inc., New York, 1977.
2. Elliot A. Tanis, "Theory of Probability and Statistics Illustrated by the Computer," Proceedings of the 1972 Conference on Computers in Undergraduate Curricula, pp. 513-20 (1972).
3. Elliot A. Tanis, "A Computer Laboratory for Mathematical Probability and Statistics," Proceedings of a Fourth Conference on Computers in the Undergraduate Curricula, pp. 416-26 (1973).

# The Computer as an Aid for Modular Instruction in Statistics

**William E. Geeslin**
*Mathematics Department*
*University of New Hampshire*
*Durham, New Hampshire 03824*
*(603) 862-2321 or 868-5639*

## Introduction

During 1973 six faculty members from six different departments at the University of New Hampshire formed a committee[1] to attempt to improve the teaching of introductory statistics. Thirty introductory courses taught by thirteen different departments were in existence and very few courses beyond the introductory level were offered. The committee suggested that a common introductory course using a modular format, video-tape, and computerized testing would have several advantages over the existing situation.

Among the advantages cited were a reduction in the number of introductory courses, improvement of student learning, student choices of statistical techniques to be learned, and freeing of faculty to teach more advanced courses. Administrators viewed this "innovation" as a possible way to lower the expense of teaching statistics to increasing numbers of students. While innovation comes slowly at the university level, the individualized course has been in existence for over a year (see Geeslin, et. al., 1975, Miaoulis, et. al., 1975, for a complete description). This paper focuses on the role of the computer in the course.

It is clear that the computer is a valuable tool for the practicing statistician. Likewise, the computer has been used increasingly for instructional purposes in a variety of disciplines. Both Computer Assisted Instruction (CAI) and Computer Managed Instruction (CMI) are enjoying widespread popularity at all educational levels. However, the majority of these efforts involve only one or two instructors in a course where the content is fixed and responsibility for the course from year to year lies with a single entity, e.g., a department or individual faculty members. The UNH course encourages variety of content and instructors. The survival of the course depends ultimately on all faculty at UNH since no department requires the new course and students enroll in the course only with the permission of their advisors.

This situation creates certain difficulties in insuring that the computer is used to the fullest advantage. That is, any functions of the computer have to be easily accessible to both module instructors and students who have little or no experience in working with a computer. Before detailing the manner in which the computer is connected with the course, a brief description of the course will be given.

The modular course consists of ten one-credit modules. Prior to registering for the course, a student consults with his advisor and determines which modules to take. Each module is approximately three and a half weeks of instruction and the normal load would be four modules per semester. Modules available are Introductory Statistics, Probability I, Probability II, Planning an Investigation I, Planning an Investigation II, Correlation and Regression I, Correlation and Regression II, Sampling I, Sampling II, and Chi-Square. With the exception of Probability II, no module requires calculus.

Two modules are under development: (1) a pre-introductory module for students having weak backgrounds in high school mathematics, and (2) a laboratory or practicum unit to provide in depth applications of the content in one of the present modules. The introductory module on descriptive statistics and statistical inference is a prerequisite for all modules except the probability modules. Otherwise, the modules can be completed in a variety of sequences. Instructors are assigned by module and come from a variety of disciplines.

During instruction the student has access to a teaching assistant for tutoring, videotapes of past instructors, catalogued instructional materials, and computer-generated examples and assignments tailored to the student's major discipline. Upon completion of each module the student is required to pass at the 80 percent level a computer-generated exam. The student may repeat similar exams (subject to announced deadlines) until he passes. The instructor of a module is allowed to determine at most 20 percent of the exam.

At the end of a semester, the student must pass a computer-generated final exam which covers the modules he completed during the semester. This exam may be repeated and is given primarily to insure retention of the material covered. The student has access to a terminal for making calculations while taking any exam.

Obviously, successful conducting of the course is heavily dependent on computer usage. In order to insure that both faculty and students can function with the computer, we rely heavily on "canned" programs. Students are required to attend a two-hour session at the beginning of the course which is designed to instruct them in using a terminal to access programs such as Statpak, SPSS, etc. A full time coordinator is assigned to the course, and is responsible for generating exams and aiding the instructor in accessing materials, conducting classroom computer demonstrations, etc. Retrieval programs that access assignments, "real" data sets, materials, and exams have been created in a manner that allows the "amateur" to use them in the same manner as a canned program. Consequently, minimal training is necessary for either students or faculty. New materials, questions, and problems can be added to the system on a continual basis.

Data banks containing questions, problems, and materials were compiled by requesting items from all U.S. and Canadian institutions offering degrees in statistics. Approximately fifty institutions responded to our requests, and as a result we have over 10,000 questions available for the exam generator. Approximately 2,000 questions (and associated answers) have been entered into the computer.

Computer retrieval of questions is used to keep manpower costs down and to allow use of a large pool of items. A classification scheme for categorizing items is used to enable easy access to the questions by a user. As test items are received they are classified by persons trained in statistics. Questions are classified according to topic, cognitive level, difficulty level, area of application, response format, need for graphing, need for a standard distribution table, mathematics prerequisite, and completion time. Each question is accompanied by an answer or answer guideline. We have found that it is possible to obtain high agreement among classifiers. Classification is somewhat complicated and time consuming. Thus we are able to collect questions much faster than we can get faculty to classify them.

The classification scheme was designed to cover the major decisions an instructor makes in determining which questions go on an exam. First we have the statistical topic(s) on which the question is based. We have identified 175 topics which might be covered in an introductory course. We have organized these topics in a hierarchical system. This hierarchy does not affect the computer program, but we feel it aids some users. We also have an alphabetical listing of topics for the user. New topics may be added to the list at any time. Many questions are classified by more than one topic if such knowledge is required to answer the question. When the item is entered into the data bank, all topics which are "above" the selected topic in the hierarchy are automatically attached to the question. Thus questions are retrievable by topic from any level of the hierarchy, but are classified according to the lowest appropriate level. Questions are stored on magnetic tape and thus the data bank is easily exportable.

Certain difficulties do occur in entering questions. Some deviations from standard notation are necessary when entering an item, e.g., "σ" must be entered SIGMA. Not all questions we have received came with answers. Thus we have to add answers which is quite time consuming. Such a wide variety of questions was received that no one person is likely to know the answers to all questions. We parcel out questions which require references and/or extensive time to produce answers. Some questions have required knowledge of the application discipline as well as statistical knowledge. We have attempted to flag these items with a warning for potential users. Also, we sometimes have to consult someone from that discipline in order to produce an answer.

Once an answer is attached to a question, the question of its correctness arises. We received some obviously wrong answers, but these cases are easy to correct. A more difficult problem arises over questions which may have different "right" answers depending on approach. For example, the classical versus Bayesian approaches. Here one can find statisticians arguing over the correct answer. Many questions ask one to test a hypothesis without specifying a technique. Several techniques may be applicable. Often the correct answer is dependent on which one of various sets of assumptions is chosen. For these questions we attempt to provide answer guidelines. The point is that providing answers to the questions does not release the instructor from being responsible for checking the answers. For use in a modular course structure, it is necessary for both instructors and teaching assistants to agree in advance concerning the "correct" approach to questions. We have found agreement can be reached with little difficulty and that this agreement can be changed as new faculty use the system.

Questions are classified by four cognitive levels: computation, comprehension, application, and analysis. The purpose of this classification is to insure that a user can guarantee he will not retrieve only computation items. Additionally, it is felt that this scheme reminds the user to both test and teach at several cognitive levels in distinction to the "recipe" course approach. Computation questions are items designed to require straight-forward manipulation of problem elements according to rules the subjects presumably have learned. Emphasis is upon performing operations, not upon deciding which operations are appropriate.

Comprehension questions are items designed to require either recall of concepts and generalization or transfer of problem elements from one mode to another. Emphasis is upon demonstrating understanding of concepts and their relationships, not upon using concepts to produce a solution. Application items are designed to require (1) recall of relevant knowledge, (2) selection of appropriate operations, and (3) performance of operations. Items are of a routine nature requiring a student to use concepts in a specific context and in a way he has presumably learned and practiced. Analysis questions are items designed to require a non-routine application of concepts.

Items are classified according to whether or not calculus is required in producing a solution. The majority of questions received do not require calculus. Questions are sorted by disciplines which provide a context or reference frame. The categories are: general, biological sciences, business, economics, education, natural sciences/technology, psychology, social sciences, and sociology. The general category is used for questions which refer to no discipline or questions on unicorns, urns, etc. with which all students should be equally familiar (or unfamiliar). Indexing questions according to discipline was used to satisfy faculty who feel it is important to teach in the context of a particular discipline. It also allows individual treatment of students according to their major.

Questions are classified from 0-9 (integers) according to difficulty. That is, a question assigned a nine would be one that 90 percent of the students in a traditional course could not answer. Another view of this is that each question is assigned a norm reference. We use the question on a criterion referenced exam, however. Also attached to the question is the estimated average completion time, i.e., how much time is required by the student to obtain an answer. The scale we use is: (up to and including) 02, 05, 10, 15, 20, 30, 45, 60, and over 60 (minutes).

Finally, the question is classified by several factors of interest only to the instructor. The required response format is one such factor. The formats we have identified are: multiple choice, essay, fill-in, numerical answer, short answer, and true/false. Some questions require either a graph that cannot be entered in the system or a standard distribution table. These questions are flagged and the instructor obtains the graph or table by hand from an indexed set. (In actual practice at UNH, we have a project coordinator who both runs the question retrieval program and obtains the graph/table.) Multiple part questions are also noted. This allows the instructor to use or avoid multiple part questions as he prefers. Usually a multiple part question will require knowledge of several topics and thus the user must examine these questions more carefully to ascertain if he has covered all the topics. Our experience to date suggests that it is useful to flag questions to particular modules. This insures that tests are appropriate for our particular course and also allows easier access to questions. Nonetheless, the system is designed so that any other statistics course can make use of it.

Materials received such as articles, handouts, and lecture notes are presently being classified in a manner similar to the test item classification scheme. These materials become available to tutors, faculty, and students to provide for both enrichment and remediation.

Our next major task (aside from continuing with the classification process) is to write the computer programs in a manner which allows use by another institution. Although we have fulfilled many requests for our materials, they are not entirely satisfactory for use on other computers. Additionally, it will take several years to complete and evaluate the classification scheme (Geeslin, 1976). Nonetheless, we feel that considerable progress has been made in providing a viable system for using the computer in a statistics course. More importantly the system is flexible enough to satisfy a variety of preferences on the part of faculty and institutions, and allow operation by the non-experts.

## Note

1. Committee members are James Cerny, Loren Cobb, William Geeslin, Richard Halley, George Miaoulis, and Jerry Warren.

## References

Geeslin, W. E. Evaluation of modular instruction. *Proceedings of the Conference on Innovation and Productivity in Higher Education*, Pittsburgh, 1976.

Geeslin, W. E., Cerny, J. W., Cobb, L. Halley, R. D., Miaoulis, G., and Warren, J. A. The University of New Hampshire statistical question retrieval facility. Report of the *American Statistical Association Study of Modular Instruction in Statistics*. Washington, D.C.: American Statistical Association, 1975, 48-52.

Miaoulis, G., Geeslin, W. E., Trainer, R. S., Cerny, J. W., Cobb, L., Halley, R. D., and Warren, J. A. The University of New Hampshire statistics teaching system. Paper presented at the annual meeting of the American Statistical Association, Atlanta, 1975.

# Effect of Computer-Delivered Testing on Achievement in a Mastery Learning Course of Study With Partial Scoring and Variable Pacing

**Richard M. Evans**
*Morningside College*
*Sioux City, Iowa*
*(712) 277-5821*

**Alvin J. Surkan**
*University of Nebraska*
*Lincoln, Nebraska*

## Abstract

The recent arrival of portable computer systems with high-level language interpreters now makes it practical to rapidly develop complex testing and scoring programs. These programs permit undergraduates access, at arbitrary times, to testing as an integral part of a mastery learning strategy. Effects of introducing the computer were studied by using, for control, data from students enrolled during the previous year. Results show that computer organized testing allowed students to take more replicates of each examination. Consequently there was, for three out of five examinations, an increase in the values of a measure of achievement based on the highest grade attained and time required before taking the final replicate desired for each examination of the course. Results of a student opinion survey indicated that in the presence of the computer organized testing, there was a greater feeling of learning than in a paper-and-pencil testing situation which allowed similar feedback with an invisible ink answer sheet. It was concluded that the feasibility of practical program development and that individual student utilization of portable, stand-alone, computer systems which now permit variable-paced mastery learning are propaedeutic in curriculum reform.

## Introduction

The purpose of this study was to introduce a time-variable competency-based (TVCB) course of study into a liberal arts teacher education program and assess its effects on student achievement and attitudes. Earlier models of school learning suggested that maximum student achievement was provided for when the time allowed for learning could vary (Carroll, 1963), and teacher activities encouraged maximum pupil activities and pursuits (Harnischfeger and Wiley, 1975). It was hypothesized that the TVCB computer testing procedure stimulates more course-related student activity, with a consequent increase in student achievement, than does a similar traditionally taught course using a paper and pencil testing procedure.

## Method

Cross-group contamination is a problem when introducing an experimental treatment into the curriculum of small schools, since most people know what others are doing. In the present case, achievement and attitude data available from a control group consisting of the previous year's educational psychology course, taught by the same instructor. Use of this control group corresponds to a version of something Tuckman (1972) calls a "Patched-Up" research design. These students had taken all five of Biehler (1974) examinations which accompany his text. They also had completed a test anxiety questionnaire (Evans, 1975), and a sixteen-question course evaluation questionnaire. The latter was designed by the instructor to record attitudes towards the course and mastery learning on a nine-point Likert-type bipolar scale.

Table 1 presents salient statistics available ascertaining the equivalence of the control group, which took educational psychology during the spring semester of 1975 and the experimental group, which took the course using the same Biehler materials one year later. Only the A.C.T. composite scores were gathered outside of the two classes. Both groups were taught the course, subject to mastery learning ground-rules. These allowed students to retake, via replicates, any of the five course examinations in which they scored lower than they may have wished. A fixed criterion arbitrarily assigned A-B-C-D-F grades to intervals with lower bounds of 90-80-70-60-<60 percent, respectively. The control group was required to achieve a score of 90 percent or more on at least four of the five examinations to obtain an "A" for the course. However, a year later the intrinsic reinforcement of computer-delivered testing allowed the instructor to require that level of achievement on five out of five examinations.

**Table 1**

**Measures Used to Equate Experimental and Control Groups**

| Group | A.C.T. Composite Score | Test Anxiety Score | Final Course Grade | Course Work Load | Time Required to Learn | Overall Course Rating |
|---|---|---|---|---|---|---|
| Experimental Group (Spring '76) | | | | | | |
| X | 21.86 | 11.76 | 3.24 | 5.30 | 4.49 | 6.78 |
| s | 5.05 | 5.76 | .80 | 2.08 | 2.16 | 2.04 |
| Control Group (Spring '75) | | | | | | |
| X | 22.76 | 12.22 | 1.96 | 5.47 | 4.82 | 6.20 |
| s | 5.12 | 6.41 | 1.11 | 1.51 | 2.01 | 2.18 |
| Univariate F (df = 1,88) | 0.67 | 0.12 | 1.74 | 0.20 | 0.57 | 1.64 |
| p< | n.s. | n.s. | n.s. | n.s. | n.s. | n.s. |

### Procedure

The treatment incorporated an IBM 5100 portable computer programmed to present replicate sets of fifteen questions randomly selected from each of the five Biehler test item pools. Students would make an appointment for a half hour out-of-class testing session. The computer was located in a carrell in a corner of the instructor's office. This made "counsel" following an examination a major aspect of the treatment. After signing on, students would be presented the randomly selected question. The question would be repeated until the examinee had identified the correct answer in an "answer until correct" procedure. After a "right" was flashed by the computer on the display screen, a feedback frame was then presented. This frame explained why one of the four multiple-choice responses was better than the other three, and referenced page numbers in the Biehler text. As the flow diagram of Figure 1 shows, students need not exit from the system until they achieve whatever mastery level their motivation allowed. The control group took their first attempt at each test in class, and

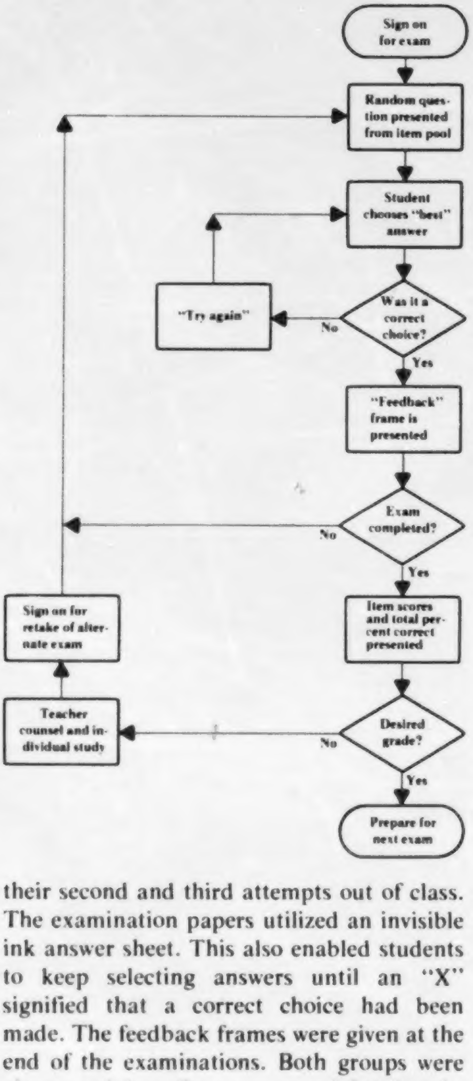

Sign on for exam
Random question presented from item pool
Student chooses "best" answer
Was it a correct choice?
No
"Try again"
Yes
"Feedback" frame is presented
Exam completed?
No
Yes
Item scores and total per-cent correct presented
Desired grade?
No
Teacher counsel and individual study
Sign on for retake of alternate exam
Yes
Prepare for next exam

**Figure 1. Morningside Mastery Strategy for Time-Variable Competency-Based Courses of Study.**

their second and third attempts out of class. The examination papers utilized an invisible ink answer sheet. This also enabled students to keep selecting answers until an "X" signified that a correct choice had been made. The feedback frames were given at the end of the examinations. Both groups were given partial credit scores on a 3-2-1-0 basis, depending on the number of choices required to obtain the correct answer was 1,2,3, or 4 respectively. This scoring has the effect of increasing the effecting duration of the examination, slightly relax the fixed percentage all-or-nothing grade criterion, and increase the test reliability (Evans and Misfeldt, 1974). Since the control group had longer tests, consisting of twenty to thirty-five questions, the obtained partial scores were adjusted to fifteen question examinations to allow contrasts with the experimental group. Only the highest score on each of the five tests was recorded for grade purposes.

## Subjects

The students were ninety sophomores enrolled during spring semester educational psychology courses in a private liberal arts college. Fifty-one students were in the 1975 (control) group and thirty-eight were in the 1976 (experimental) group. The numbers of these enrolled in music, elementary education, and secondary education were 11, 13, 27, and 5, 10, 23 in the two groups (Chi square = 1.09, df = 2, not significant). The groups did not differ significantly in A.C.T. composite scores, test anxiety scores, or final course grades in educational psychology. Since the testing method used in the course enabled most students to establish their expected course grades prior to the last week of the semester, three questions from the attitude measure were deemed important for equating the groups. They were: (1) Student perception of the course work load, (2) the time required for the students to learn compared to their perceptions of the amount required by other students, and (3) overall rating of the educational psychology course. As Table 1 shows, the groups did not differ on any of these measures.

## Results

Measures opposite to the hypothesis were (a) scores on the five course examinations, (b) the number of attempts students made in order to achieve their particular course grade objective, and (c) student perception of the amount learned during the course. A total of 371 sets of examination questions were answered by the control group and 385 by the experimental group. Each student took an average of 7.27 and 9.87 examinations, respectively for each course. All measures were asceded univariate analysis of variance tests with a .05 level criterion for significance. As Table 2 reveals, the experimental group

Table 2

Measures Used to Determine Differences Between Experimental and Control Groups

| Group | *No. Tries First Exam* | *No. Tries Second Exam* | *No. Tries Third Exam* | *No. Tries Fourth Exam* | *No. Tries Fifth Exam* | *Perception of Amount Earned* | *Scores Test One* | *Scores Test Two* | *Scores Test Three* | *Scores Test Four* | *Scores Test Five* |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Experimental Group M=38 | | | | | | | | | | | |
| $\overline{X}$ | 1.87 | 1.90 | 2.36 | 2.15 | 1.59 | 7.16 | 40.82 | 40.85 | 41.08 | 39.13 | 39.13 |
| s | .77 | .85 | .84 | 1.23 | .68 | 1.66 | 2.67 | 2.72 | 8.76 | 3.73 | 3.47 |
| Control Group M = 51 | | | | | | | | | | | |
| $\overline{X}$ | 1.41 | 1.35 | 1.80 | 1.57 | 1.14 | 5.80 | 40.45 | 38.55 | 37.65 | 38.10 | 37.51 |
| s | .61 | .59 | .78 | .78 | .35 | 2.31 | 2.86 | 4.35 | 4.08 | 10.43 | 4.30 |
| Univariate F | | | | | | | | | | | |
| (df=1,88) | 10.11 | 12.75 | 10.51 | 7.60 | 16.96 | 9.32 | .39 | 9.84 | 6.10 | 0.35 | 4.56 |
| p | .01 | .01 | .01 | .01 | .01 | .01 | n.s. | .01 | .02 | n.s. | .05 |

took significantly more of each of all five examinations, and scored higher on three of them. This group also indicated learning more during the course than in regular courses relative to the control group.

The variable-pacing aspects of the experimental group were examined according to the number of 113 available days which elapsed between the semester beginning that each student achieved his target criterion level on each examination. Table 3 presents mean and standard deviations of these elapsed times for each examination. Most notable is the increased time variance near the middle or third examination, which was the most difficult experienced by the students.

Table 3

Mean and Standard Deviation of Days Taken Into Semester (Experimental Only)

| | *First Exam* | *Second Exam* | *Third Exam* | *Fourth Exam* | *Fifth Exam* |
|---|---|---|---|---|---|
| $\overline{X}$ | 25.59 | 43.85 | 73.31 | 89.74 | 101.41 |
| s | 9.83 | 15.79 | 22.02 | 15.68 | 11.40 |

## Discussion

The higher achievement in three of the course examinations by students conducted with the computer-organized procedure support Bloom's suggestion that most students can achieve at a level heretofore attainable only by the best students (Block, 1971). This higher achievement appears to be related to increased student activity outside of class, as shown by the increase in examinations taken. This being so, the key element of the Harnischfeger-Wiley model (1975), which asserts that activities which most affect "student pursuits" have the greatest direct effect on student achievement, is also supported by the present findings. An earlier model for school learning (Carroll, 1962) suggested that if mastery were to be held constant, students would require variable periods of time to achieve that mastery. The consistent increases in standard deviations of elapsed time for achievement of the students' particular grade criterion until the third test, followed by a decrease in standard deviations of tests four and five, suggest however, that the present treatment did still not produce a completely time variable course of study. The constraints of the college's traditional fixed-time, variable-mastery curriculum required that after 113 days the semester end and letter grades be submitted to the registrar. Two students who had not taken all five examinations by that time were given no credit for the course.

The "threat" of no credit and the requirement to sign up for the course in another semester in order to complete its activities seemed to produce a flurry of activity during the last few weeks of the semester, possibly causing the decrease in standard deviation of lapsed time in taking the last two examinations. In this sense, one of the major obstacles to a "pure" mastery learning strategy—the fixed time units in the semester—was assaulted on an individual course basis, but not yet conquered because the remainder of the college curriculum shaped student attitudes toward finishing when other courses ended, rather than opting to repeat segments of the course not completed at the 90 percent level.

The increase in examination scores without a corresponding increase in course grades should be mentioned. Again, the instructor felt that the experimental treatment allowed him to increase the grade requirement to five out of five for a particular grade. This increase in stringency was not felt possible using the paper-and--pencil testing procedure.

The present findings are in agreement with a study by Cartwright and Derevensky (1975) who also used computer-organized testing with the Biehler educational psychology materials. Their attitudinal study found that students perceived this to be an effective method to incorporate in mastery of course material.

It should be noted that the feedback following each of the multiple-choice questions and the counsel by the instructor or his assistant following examinations were an important part of the experimental treatment. Such human contacts tended to personalize what could become a rather mechanistic, impersonal, and private method of individually progressing through the material.

Indeed, Block and Burns (1976) have attempted to show the emphasis of mastery learning on the social aspects of education over the relatively private individually prescribed instruction. Office hours during the experimental period were increased by a factor of four in order to discuss test questions and other educational psychology subject matter. A significant portion of this time was spent trying to convince some students that they did have the ability to score well on these tests, and to keep trying. The immediate feedback often revealed that students were missing material which they said they knew quite well. This seemed to be due to poor reading habits, which improved as they acquired *savoir faire* in taking tests. Other students remarked that they enjoyed the text book more than had others. It was conjectured that these students, for perhaps the first time in their academic careers, were required to read technical material to mastery, which in turn brought pleasure. Finally, the social aspects of the treatment in both groups was enjoined by the requirement that all students attend all class discussions, no matter how far they had progressed individually through the material.

## Conclusion

With the arrival of portable computers with high-level language interpreters, small schools can now enter the computer-assisted instruction arena, which has in the past been limited to larger universities possessing the large computers. The time-variable-competency-based course using the computer-delivered testing allowed by low cost portable computers shows promise in providing methods for keeping track of a multitude of individual student differences in time required for learning. This is seen as the greatest obstacle to implementation of pure forms of mastery learning in the schools. If the basic theory of mastery learning is sound. then the time has come that we reorganize schools to be in accord with this approach rather than attempt to modify our mastery learning strategies to fit into the existing school systems. The results of the present experiment have demonstrated that this time is at hand since the technology is already available.

## References

Biehler. R. *Instructor's Manual to Accompany Psychology Applied to Teaching*. Boston: Houghton Mifflin Co.. 1974.

Block. J. H. *Mastery Learning. Theory and Practice*. N.Y.: Holt. Rinehart and Winston. Inc.. 1971.

Block. H. H. and Burns. R. B. Mastery Learning. Mimeographed. University of California. Santa Barbara. 1976.

Carroll. J. B. "A Model for School Learning." *Teachers College Record*. 1963. 64. 723-733.

Cartwright. G. F.. and Derevensky. J. L. An Attitudinal Study of Computer Assisted Testing as a Learning Method. Paper presented to the 59th Annual Meeting of the American Educational Research Association. Washington. D.C.. April 1975.

Evans. R. M. Confidential Questionnaire Concerning Education 215. Mimeographed. Morningside College. Sioux City. Iowa. 1975.

Evans. R. M.. and Misfeldt. K. Effect of self-scoring procedures on test reliability. *Perceptual and Motor Skills*. 1974. 38. 1248.

Harnischfeger. A.. and Wiley. D. E. "Teaching-Learning Processes in Elementary School: A Synoptic View." *Studies of Educative Processes*. No. 9. University of Chicago. 1975.

Tuckman. B. W. *Conducting Educational Research*. New York: Harcourt. Brace. Jovanovich. Inc. 1972.

# Maximizing the Transferability and Utility of A Computer-Assisted Test Construction System

**P. L. Olympia, Jr.**
*Department of Chemistry*
*University of the District of Columbia. Van Ness Campus*
*4200 Connecticut Avenue. N.W.*
*Washington. D.C. 20008*
*(202) 282-7377*

## Introduction

In computer-assisted test construction (CATC). as in other areas involving technological innovations. academics have unfailingly demonstrated their proclivity for reinventing the proverbial wheel. In the summer of 1975. G. Lippey. one of the early CATC devotees. listed more than ninety CATC systems in one form or another.[1] and the list appears to continue to grow rapidly. Unfortunately. a great number of CATC programs have come and gone over the years because they were primarily localized (some say. "Band-Aid") operations. If one lesson were to be learned from the forced abortion of the 1976 CATC conference (for lack of registration) it may well be that CATC cannot survive on an *ad hoc* basis. The continued survival of a given CATC system depends. for the most part. on its ease of use. algorithmic efficiency. cost-effectiveness. transferability and. perhaps ironically. other uses beyond the mere automation of testmaking.

Most. if not all. CATC packages do not require any significant knowledge of computer hardware or software on the part of the user. Yet. it is naturally desirable for a user to be able to create a test according to his specifications. Unless one has designed an easy-to-use test item classification system. a user can easily get lost in a quagmire of item indices. cross-references. etc.

The transferability of a typical CATC system ideally should apply to both the test-generator program and the test item data bank. This usually means a higher-level programming language of widespread use. e.g.. FORTRAN. and again. a coherent test item classification system. Although many CATC authors are reluctant to share their test generator programs. they are generally willing to share their data banks. This is fortunate because the cost of developing a "good" data bank far outweighs that of developing a test generator program regardless of its apparent sophistication. In the SOCRATES system. for example. it is estimated that each banked test item costs between five and ten dollars.[2]

This report describes a test item classification system which recently was implemented in the CATC portion of a CMI package called TITA2 (Totally Iterative Testing and Analysis. version 2) originated at the University of the District of Columbia and distributed to approximately thirty colleges world-wide. This classification system holds the promise of facilitating the exchange of test item data banks among CATC users irrespective of the nature of their test generator programs. This report also describes how such test item classification systems produced a number of side benefits in the optimization of learning and instruction.

## Background

The TITA2 system is a computer-managed instruction package. written in Extended BASIC. at the heart of which is an efficient algorithm for the computer-generation of repeatable examinations.[3] The versatility of the algorithm results from the fact that test items are banked in disjointed phrases called the "keyphrase," "statementphrase," and "distractorphrase," so that the computer has the freedom to assemble related phrases in any one of four objective item formats to form logical test questions. Each "keyphrase" (not test item since none exists in the bank, *per se*) is banked with a non-unique classification number that contains information concerning the behavioral objective a generated test item incorporating the keyphrase is supposed to measure.

### Test Item Classification System

Many test item classification systems presently in use rely upon arbitrarily selected code numbers for subject matter and difficulty level. This approach appears to be unwise because (1) in reality, every test item, regardless of whether it is intended for inclusion in a norm-referenced or criterion-referenced test, exists for the purpose of measuring the attainment of a given educational objective. Hence, it appears that ambiguities can be prevented by using a classification system principally based, not upon subject matter and difficulty level, but upon specific, measurable behavioral objects; (2) if one were intent upon incorporating subject matter and difficulty level designations in the classification system, the arbitrary selection of code numbers appears to be totally unnecessary since there already exist widely accepted code numbers for these. One can just as easily use a Dewey Decimal Classification (DDC) number[4] for the subject matter and a Bloom taxonomy level[5] to indicate difficulty.

In the 1,000 divisions of the DDC a subject matter is given a three digit number, e.g., 574 for biology, 539 for modern physics and 612 for human physiology. Although the entire Dewey system is far from being static, it is true, nonetheless, that the 1,000 code designations are relatively stable. Conceivably, we could have chosen the Library of Congress notation in place of the DDC. However, for computational expediency, straight numeric data must be preferred over alphanumeric data.

In Bloom's taxonomy a behavioral objective in the cognitive domain falls into one of six general classes: (1) knowledge, (2) comprehension, (3) application, (4) analysis, (5) synthesis, or (6) evaluation. Now, a behavioral objective is often considered to be either terminal or enabling. Normally, one or more enabling objectives lead to a given terminal objective, such that an enabling objective cannot have a taxonomy level higher than the terminal objective to which it leads.

From these considerations, then, a test item classification number in TITA2 consists of a nine-digit number where the last three digits are separated from the first six by a decimal point placed there as a programming convenience as well as for readability. That is,

AAABBB.CCD

where, AAA is the Dewey code in the 1,000 divisions, BBB is the terminal objective number; CC is the enabling objective number; and D is the taxonomy level. Thus, a test item ("keyphrase") with a classification number of 546025.314 is one from inorganic chemistry (546) designed to assess the attainment of enabling objective 31 belonging to terminal objective 025. The highest cognitive process involved for this item is one of analysis (level 4).

Those who were introduced to TITA2 initially balked at the use of the taxonomy level digit. They became more comfortable with it after realizing that the taxonomy level: (1) serves a useful purpose in validating course content. That is, a test item bank consisting of items with taxonomy levels of mostly one and two (except unavoidably in some areas such as human anatomy) tends to encourage not much more than simple recall and rote learning; (2) facilitates the sequencing of learning tasks by virtue of the hierarchy built into the taxonomy.

What this test item classification system implies is that one should be discouraged from banking a test item without having a clear idea as to the behavioral objective the item is intended to measure. Unfortunately, here, one runs into an old problem faced by producers of programmed instruction materials—the lack of any extensive or workable compilation of behavioral objectives for the multitude of academic disciplines, leading Short to say:

> I smile every time I see that someone else has suggested that a teacher. . . should choose a program by "comparing his course objectives with the program's objectives." I can only think of the poor teacher . . . comparing two blank pages, one *not* listing his own educational objectives in behavioral terms and the other *not* listing the programs.[6]

Short suggests that the dearth of published objectives may well be due to the sad fact that the "behavior of writing objectives is seldom followed by anything resembling reinforcement."

In part to remedy the present situation and in particular to demonstrate the viability of the procedure suggested here, we expect to publish during the next two years a useful list of objectives in mathematics, chemistry and biology, and possibly also in English and physics, including, of course, a coherent numbering system for most terminal and enabling objectives in these areas. In the meantime, CATC users can still exchange test item banks within the context of the procedure suggested here if those banks are accompanied by a bank of behavioral objectives.

At first, it may seem that the development of a bank of objectives may easily double the cost of a CATC system. In reality, the additional cost is relatively insignificant because if an objective is properly written, one discovers that, in many cases, there is little to distinguish the objective from its test items. Surely, whatever additional cost the development of a bank of objectives may entail can be justified by the advantages described in the next section.

One obvious advantage of an item classification system such as the one described here is that it allows for a straightforward test specification procedure using test item attributes with which the average teacher is familiar. Since a user can retrieve items by citing as specific an attribute as an enabling objective, it also avoids the necessity of a user rejecting a retrieved item—a not uncommon occurrence in item selection via general topical heading. In this manner, it has the same advantage as having a catalog of test items without many of its disadvantages.

Surely test item classification using behavioral objectives as one attribute is not new. What is new is the coupling of a simplified, readily-learned, four-tier classification index with an efficient test generating algorithm. TITA2 produces any number of equivalent tests within a short time span which is essentially independent of the size of the original data bank. This is accomplished, just prior to test generation, by the system's ability to create a renumbered and repacked subset of the original data bank containing only those potential test items that satisfy the characteristics specified. The randomized search for related test item phrases is then confined only to this subset. As the data bank undergoes revision, the number of "key-phrases" that satisfy a given item characteristic naturally changes. When the system creates the data bank subset, this information is gathered and given to the user so that he knows how many potential test items containing the desired characteristics are in the bank. Thus, even if the data bank is revised daily, the user is assured of getting up-to-date information concerning the bank.

If a user does not have a list of desired objectives, TITA2 will print one for him. He may generate a list of objectives by indicating any combination of the four classification categories, he may also generate a list within a given DDC number by indicating index terms such as "pH." Since TITA2 not only formulates test items itself but also creates the data bank by prompting the terminal user for appropriate information, the entire test-making operation is essentially machine-directed. Perhaps such a system should be considered to be an example, not of CATC, but of MATC—Man-Assisted Test Construction system (or, in deference to the current rights movement, PATC).

## Other Benefits

We are gratified by the fact that what, for us, began as an attempt to standardize input data to a test-construction system turned out to have many applications in systematizing learning and instruction.

First, the TITA2 system, along with its bank of objectives, has proved indispensable in the writing of instructional modules and study units within the framework of modular instruction. The following sections of a typical instructional module are produced directly from TITA2: (1) list of prerequisite objectives, (2) list of module objectives, (3) readiness (inventory) test, (4) pretest, (5) post-test, and (6) test key portion of the teachers' guide. The taxonomy level of an objective has helped in the derivation of objectives dependency networks and the concomitant determination of modular prerequisites. The repeatability of tests generated by TITA2 has facilitated the production of equivalent instructional modules and also the assessment of the effectiveness of alternate learning activity sequences prescribed in the module.

Second, graded papers of students now list the items they missed, the correct answers to these items, the objectives in which they are deficient, and in most cases, alternate learning resources which cover these objectives.

Third, the bank of objectives permitted the automated production of course outlines (which, as a matter of school policy, are given to students on the first day of class) and the catalog description of course offerings. Clearly, this system also allows for rapid revisions of course outlines and catalog descriptions as the needs arise—for example, our switch from a quarter to a semester schedule. The bank of objectives and the generated course outlines also should prove useful in the granting of credits to transfer students.

Finally, TITA2 provided an important impetus for reexamining the adequacy and effectiveness of current holdings of the instructional resources center. The trick is to match the objectives with available resources. This activity has served to identify irrelevant and obsolescent media materials and has produced an inventory of needed resources.

## References

1. G. Lippey, personal communication. See also: G. Lippey, ed. *Computer-Assisted Test Construction*, Educational Technology Publications, N.J. (1974).
2. O. Seeley, Jr. and W. V. Willis, *CCUC/6 Proceedings*, 135 (1975).
3. P. L. Olympia, Jr., *Educational Technology*, 15 (6), 53 (1975); *CCUC/6 Proceedings*, 155 (1975).
4. M. Dewey, *Abridged Dewey Decimal Classification*, 10th ed., Forest Press, Inc., Lake Placid (1971).
5. B. Bloom, *Taxonomy of Educational Objectives, Handbook I*, Longmans, Green and Co., N.Y. (1956).
6. J. Short in *Trends in Programmed Instruction*, G. Ofiesh and W. Meierhenry, eds., National Education Association, Washington, D.C., 162 (1964).

# Computer Processed Evaluation

**George H. Griswold[1] and George H. Kapp**
*Chemistry Department*
*Washtenaw Community College*
*Ann Arbor, Michigan 48106*
*(313) 971-6300*

## Abstract

This paper deals with a system of computer generated and scored equivalent, but unique repeatable tests based on performance objectives, containing varying numbers of multiple format questions which provide for immediate feedback to students on a cost effective basis, which was developed in the Department of Chemistry at Washtenaw Community College, Ann Arbor, Michigan.

## Introduction

Faced with a constantly expanding student population and all of the problems which are associated with it, the Department of Chemistry at Washtenaw Community College has turned to the development and use of a systems approach utilizing the computer for those tasks which are objective and clerical in nature. The system identified four key parts: (1) objectives, to formally set the goals to be achieved, (2) the instructor, to interpret the objectives to the student, (3) the student to receive the instruction, and (4) evaluation to provide the feedback for the other parts of the system. The primary use of the computer is in the evaluation part of the system where it releases the instructor from the clerical routine of generating and scoring test instruments, and permits him to concentrate on the instructional part of the educational process.

The cornerstones of our system are the sets of performance objectives, thus the other parts of the system are keyed to them. At the beginning of each course, the students are informed that the objectives for each unit of study will be duplicated and distributed. The students are then encouraged to use the objectives in a "checkoff" manner as each unit progresses, as well as in their final review prior to taking the test on that unit. Further, it is made clear to the students that all of the questions will be drawn directly from the objectives. With the students keyed in on the objectives, the instructor and the evaluation systems must also be keyed to the objectives.

## System Components

The evaluation part of our system consists of four separate programs written in FORTRAN IV: (1) a LOADER which reformats the test questions, (2) a TEST GENERATOR which creates tests according to a given set of specifications, (3) a GRADER which evaluates student responses and records student grades, and (4) a GRADEBOOK for editing and summarizing student records. Using these four programs on the University of Michigan Terminal System (MTS) we have been able to implement the evaluation part of our system for two courses—Introductory Chemistry 057, and General Chemistry III.

In addition to the four operating programs, our system includes the questions which are associated with the seventeen units divided between the two implemented courses. Our test questions are each keyed by objective and unit, and maintained in machine readable form with each of the seventeen units being held in a separate file. The questions were originally entered into disk files using a terminal to take advantage of the special and lower case characters which are available on the IMB TN print chain which was to be used for output of the tests. The number of questions in each unit's file (bank) varies with the number of objectives in the unit, and the number of questions (versions) attached to each objective. While the number of versions varies from one to nine—the maximum permitted—most objectives currently in our banks have five questions attached. Because of the large volume of questions involved in the seventeen units, they have been copied from their original location on disk files to magnetic tape for back-up and storage purposes. While the question banks are on disk file, they are usually edited using the MTS EDITOR. In this way the banks are maintained in a current and "living" condition, which we feel has been vital to the success of our system.

## Test Generation

The way in which our TEST GENERATOR program selects questions for tests is based on a different philosophy than other systems which were reported at the last CATC Conference. In establishing a set of performance objectives and distributing them to the students all of those things which the student will be expected to be able to do have been established. Thus, an evaluation of those items should in theory be attempted on a test. However, in dealing with a classroom situation where time is a limiting factor, it is not possible to test each student on all of the objectives. Therefore, rather than make a subjective decision as to which objectives are "more important" than others, our TEST GENERATOR has been programmed to select questions for a given test form by first randomly selecting an objective, and then randomly a question attached to that objective (version), without selecting the same objective twice for any one test form. By having the test questions selected in this manner, two things are accomplished: (1) The student is forced to study all of the objectives which promotes more learning. (2) The probability of student complaints as to the content which was covered by the test are greatly reduced. Using our system the only subjective decision the instructor need make is whether a particular question should be placed in the bank and attached to a specific objective.

Another unique feature of our TEST GENERATOR is its ability to accept questions of more than one type. Formally, the TEST GENERATOR is programmed to accept four types of questions: (1) Fixed answer, (2) multiple choice, (3) random number, and (4) double random number. The fixed answer type of question can be used for: true-false, multiple true-false, multiple choice, matching, fill-in, and key word formats when the proper wording and instructions are used with the question. This type of question is limited at the present time to six responses, but could easily be expanded to nine.

**Figure 1**

A Fixed Answer Question

Select "O" for oxidation, or "R" for reduction, as the name of the process described below:

A loss of electrons.

The multiple choice type of question is used for questions having one answer and four distractors of not more than thirty characters each. As the test forms are generated, the answer and distractors are rotated randomly among the referencing letters (A) through (E), while the identity of the answer's position is maintained for grading purposes. When either the answer of the distractors require more then thirty characters, multiple versions of the fixed answer type can be used to accomplish the same result.

**Figure 2**

A Multiple Choice Question

Select one of the following as the oxidation number for the ion shown below:

$Na^{1+}$

(A) +2
(B) 0
(C) +1
(D) +3
(E) −1

The random number type is used for calculation problems whenever multiple versions of the same type of calculation are desired. The problem is worded in a way as to reference the random number which is printed below the problem according to a given set of specifications including: form (decimal, or scientific), value, number of digits, and unit (limited to one line of thirty characters). The answer is calculated using a given set of specifications including: the mathematical equivalent of all of the other values in the problem, and whether to multiply, or divide this value into the random number, saving the answer for grading purposes.

**Figure 3**

A Random Number Question

Calculate the number of grams in the number of milligrams shown below:

Report the first digit to the right of the decimal point in your answer.

6812.1 milligrams

The double random number type is used for questions involving the simple mathematical operations of addition, subtraction, multiplication (also squaring), and division of two numbers. The problem is worded in such a way as to reference the two random numbers which will be printed, each enclosed in parenthesis with a specified mathematical sign between them, below the question, according to specifications including: form (decimal, or scientific), value, number of digits, and with the first number equal to, less than, or greater than the second number. The answer to the problem is calculated using the specified operation, and maintained for grading purposes.

**Figure 4**

A Double Random Number Question

Calculate the answer to the problem below:
Rewrite your answer in standard exponential form if necessary, and then report, in order:

a. The first digit to the left of the decimal point in your answer.

b. The sign of your exponent, using "P" for +, or "N" for −.

c. The exponent of your answer.

$(4.5743 \times 10^{+06}) + 2.5297 \times 10^{+03})$

As we were interested in producing repeatable tests, our TEST GENERATOR was programmed to create test forms of equal difficulty, but not necessarily equal numbers of questions. By generating test forms of equal total difficulty, but with varying numbers of questions we have been able to almost completely eliminate any security problem created by the release of old test forms. Using this concept any one test form may contain a large number of "easy" questions, a small number of "difficult" questions, or some random mix of "easy" and "difficult" questions. To obtain the proper number of questions of the right difficulty for each test form, each question is assigned a number of points subjectively to reflect its difficulty. One point is assigned to those questions which are considered to be "easy." Two, three, or more points may be assigned to more difficult questions. the assignment of additional points to a question usually anticipates that it will take an average student two, three or more times longer to consider, calculate, or otherwise develop a correct response to that question. With each available question having been assigned a number of difficulty points, the TEST GENERATOR can select questions randomly according to objective and version until a specified number of points is reached to complete each test form, thus creating unique tests with equivalent difficulty which are repeatable and present a minimum security problem.

The actual process of test generation using our system is extremely simple—after the questions have been written, assigned to an objective, given a number of difficulty points, and placed in a machine readable form on disk file. Using the CONTROL feature of the MTS system which premits batch jobs to be queued for off hours processing from a terminal, we can initiate the generation of any number of completely unique test forms in a terminal session which lasts no more than five minutes. To generate the tests the LOADER program must be run to reformat the question bank. Normally the only output produced at the terminal by the LOADER is a statement of the total number of objectives in the bank being used. (This information is useful as the TEST GENERATOR was programmed to limit the size of each test form to a number of questions which is 80 percent of the number of objectives in the bank.) However, if problems in reformating the bank are detected specific error messages will be displayed. Once the bank has been reformatted, the TEST GENERATOR can be run, and the four specifications necessary entered in order: (1) the number of tests to be generated, (2) the total number of points for each test form, (3) the first line of the test title, and (4) the second line of the test title. Following the entry of these specifications there is little to do until the test forms are ready to be picked up from the line printer.

The printed test forms themselves contain several notable features. The most notable is the printing of the answer(s) to each question on the right-hand side of the paper. A heading, is printed at the beginning of each test form on both the test and answer sides of the paper containing: the test form number, and the test title. To aid in the separation of the answers from the test form, a cut mark is printed in each line of printing. When the answers are separated from the test forms, the test forms take on the usual 8½ by 11 format. The format used for each question includes the printing of the question number, number of points assigned to the question, and the response number(s) (position on the answer sheet) for the question. Following the basic information, the instructions, text, and any required random number(s) are printed. Another notable feature of the printed test forms is the separation of questions between pages. (Unless a question exceeds one page in length—the LOADER program will accept questions with up to ninety-nine lines of text.) If the printing of the next question can not be completed on the current page, the rest of that page is skipped and the printing is continued at the top of the next page. To aid in the separation of one test form from another, a line of printing is placed on the crease between the bottom of one test form and the top of the next test form. Through the use of these notable features it is possible to print test forms and answers which contain all the essential information and are yet readily separable into tests and answers.

### Use and Scoring of Tests

In our system the tests are administered during a regularly scheduled class period of fifty-five minutes. The students are expected to have available at these sessions a hand-held electronic calculator and a pencil for marking the answer sheet. As each test form is unique, the test forms are given out one at a time with an answer sheet to prevent those students having the belief that one test form may be easier than another from sorting through the stack and holding up the distribution process. The students are encouraged to set-up any calculations on their test form and then use their calculator to obtain the answers, writing them down directly under where the response number(s) are shown on the test form. Only after all of the questions have been answered in this manner, is the student encouraged to fill out the answer sheet. When the student has completed and checked the answer sheet, it is checked again by the instructor to verify that the student has recorded an identification number, test form number, and that the number of answers given meets or exceeds the number of points assigned to that test. Immediately, after checking the answer sheet, the student is handed the answers to the test form indicated on the answer sheet. (The student always retains the test form.) Through this procedure, the student is immediately able to score the test by comparing the answers with those responses previously recorded on the test form, and determine the percentage score which will be recorded later when the answer sheets are processed. Currently we offer each student two additional opportunities to take a test on the same unit of study during prescheduled periods outside of the regular class meeting times. This procedure of test administration eliminates most of the need to evaluate excuses for absenteeism on test days, and has the added benefit of providing a mechanism whereby the tests can be used as a learning tool.

In addition to the generation of tests, our system also scores and records the grades obtained by the students. As the TEST GENERATOR creates the test forms, it stores the answers in a disk file. The scoring process starts with the conversion of the answer sheets into machine readable form (currently punched cards from an optical scanner). The GRADER reformats the answers on the disk file and then compares them sequentially with the student responses on a match or mismatch basis. One point is given for a match, and no points for a mismatch. The points earned are then divided by the total points possible and a percentage calculated. This percentage is then recorded in the disk file where the student records are kept. With the repeatable feature of the test forms, we have programmed the GRADER to retain only the highest percentage obtained on each of the test forms of a given test taken by a given student. Thus under this system the student has nothing to lose by repeating a test. After the grade has been recorded, a grade report is generated which gives the test form number, the student's identification number, test title, question numbers, response numbers, the student's responses, and the answers. In addition the report shows the points earned, the total number of points possible, and the percentage. This report is distributed to the student so that he/she can verify the questions which he/she missed, and the grade which was obtained. The GRADER also produces a summary report for posting. The summary report shows the test title, the date on which the tests were scored, the average percentage grade for all the tests scored, and a listing of the identification number of each individual and corresponding percentage grade, sorted by identification number. With this report, the students can determine how they did on all of the tests they took, whether they improved or not, and how their tests compared to all of the tests which were scored. The GRADER also produces a third type of report which shows the total number of responses which were expected for each question in the bank for that test, and the number of correct responses which were received. This report is useful in evaluating how randomly the TEST GENERATOR is selecting questions. This report can also be used to evaluate whether a question is too difficult, faulty in some other manner, or whether the instructor failed to present the material in an effective manner. Through the use of the GRADER we are able to score and record the grades for a large number of unique tests, and produce three different kinds of reports which permit further evaluation, and feedback into our system.

In special cases we do hand-grade a few tests using the answer side of the printed form before giving it to the student. The problem of recording the grades from these hand-graded tests prompted the development of our GRADEBOOK program which gives us the capability to edit the student record file. With the GRADEBOOK we can change a student's identification number, name, or section number as well as any of the grades obtained by that student. Using the GRADEBOOK we can also add or delete complete student records to the student record file. An additional feature of the GRADEBOOK is its ability to permit a complete set of grades to be added to all of the records in the student record file when a book report or other non-test grade is part of the course. However, the most useful feature of the GRADEBOOK is its ability to compute weighted percentages using the recorded grades for each student at the end of a marking period. Through the use of the GRADEBOOK, an instructor can modify a student's grades to reflect any special circumstances which may occur, and also compute accurately and rapidly course grades at the end of a marking period according to the instructor's subjective evaluation as to how each grade should be weighted relative to the total course grade.

## Economics and Summary

This past fall semester we generated over 3,000 tests having over 50,000 questions for the students in the two courses. With three opportunities to take each test, we found that it was necessary to generate a number of tests which was 1.5 times the number of students active in the course. Since we experienced a change in the rate structure at MTS during the latter part of the semester (a more favorable rate was implemented for those willing to process later at night), it is difficult to give any exact cost figures for the entire semester. However, under the new rates, our cost to generate each question appears to be on the order of .6 cents depending somewhat on the number of text lines which have to be printed. Our cost of scoring, and recording of the tests is on the order of .1 cents per question when the tests are processed in batches of 200 or more, but does not exceed .2 cents per question when as few as ten tests are processed. As our answer sheets permit no more than twenty-eight responses, the expense involved in generating and scoring a test of twenty-five responses (the maximum we use) is on the order of 25 cents when all of the additional costs have been figured. Based on these figures and the educational gains which are possible using this system, we feel that it has proven to be cost effective.

## Note

1. All correspondence with regard to the contents of this paper should be directed to Professor Griswold.

FORM NUMBER 2

CHEMISTRY 57
TEST NO. 3

NAME ___________

1. (3 POINTS)

Calculate the answer to the problem below:
Rewrite your answer in standard exponential form if necessary, and then report, in order:

a. The first digit to the left of the decimal point in your answer.
b. The sign of your exponent, using "P" for +, or "N" for - .
c. The exponent of your answer.

(9.9205 X 10+06) + (8.3701 X 10+03)

RESPONSES( 1- 3)

2. (1 POINT)

Convert the following number to decimal form.
Report the first digit to the left of the decimal point in your answer.

3298.9 X 10+00

RESPONSE( 4)

3. (1 POINT)

Select one of the following as the way in which 2.56 should be written in standard exponential form.

| | |
|---|---|
| (A) 2.56 X $10^0$ | (D) 2.56 X 1 |
| (B) none of them | (E) 2.56 |
| (C) 256.0 | |

RESPONSE( 5)

4. (1 POINT)

Write the following number in standard exponential form.
Report the first digit to the right of the decimal in your answer.

209.18

RESPONSE( 6)

FORM NO. 2

CHEMISTRY 57
TEST NO. 3

1.
9.92885E+06

2.
3298.908

3.
(A) 2.56 X $10^0$

4.
2.09177E+02

```
FORM NO.    57                                          WARD CHERYL D
                                 CHEMISTRY C57          JOURNAL NO.   404
                                  TEST NO.   8

                   QUESTION   RESPONSE    REPORTED    CORRECT    POINTS
                    NUMBER     NUMBER      ANSWER     ANSWER

                       1          1          A          A          1
                       2          2          4          4          1
                       3          3          T          T          1
                       4          4          3          1          **
                                  5          3          3          1
                                  6          2          2          1
                       5          7          B          B          1
                       6          8          2          2          1
                                  9          8          8          1
                       7         10          T          T          1
                       8         11          F          F          1
                       9         12          3          3          1
                      10         13          3          5          **
                                 14          1          4          **
                                 15          C          6          **



                                                    TOTALS POINTS =   11

                                                  POINTS POSSIBLE =   15

                                                      PERCENTAGE =  73
```

GRADE REPORT

CHEMISTRY [redacted]     TEST NO. 5     DATE DEC 10, 1976

NUMBER OF TESTS GRADED = 71

AVERAGE PERCENTAGE FOR ALL TESTS = 73.500

| JOURNAL # | PERCENT | JOURNAL # | PERCENT | JOURNAL # | PERCENT | JOURNAL # | PERCENT | JOURNAL # | PERCENT |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 64 | 35 | 84 | 83 | 84 | 111 | 68 | | |
| 3 | 48 | 37 | 72 | 83 | 84 | 111 | 56 | | |
| 5 | 84 | 39 | 72 | 84 | 84 | 113 | 48 | | |
| 9 | 96 | 39 | 96 | 84 | 93 | 116 | 64 | | |
| 9 | 92 | 44 | 81 | 86 | 88 | 116 | 100 | | |
| 12 | 81 | 44 | 92 | 86 | 88 | 117 | 88 | | |
| 16 | 80 | 49 | 72 | 89 | 52 | 118 | 62 | | |
| 18 | 81 | 50 | 56 | 90 | 38 | 118 | 88 | | |
| 18 | 76 | 53 | 68 | 90 | 48 | 119 | 72 | | |
| 19 | 48 | 53 | 96 | 92 | 44 | 119 | 84 | | |
| 19 | 60 | 55 | 60 | 92 | 58 | 448 | 89 | | |
| 20 | 92 | 58 | 96 | 93 | 84 | | | | |
| 21 | 60 | 60 | 80 | 94 | 40 | | | | |
| 21 | 88 | 66 | 76 | 94 | 68 | | | | |
| 23 | 84 | 69 | 80 | 101 | 96 | | | | |
| 24 | 76 | 74 | 68 | 103 | 56 | | | | |
| 24 | 73 | 80 | 38 | 103 | 84 | | | | |
| 29 | 68 | 80 | 72 | 107 | 64 | | | | |
| 29 | 72 | 82 | 56 | 110 | 80 | | | | |
| 33 | 48 | 82 | 60 | 110 | 85 | | | | |

```
   RESPONSE DISTRIBUTION               CHEMISTRY 057

         TEST NO.    1            DATE  SEP 23, 1976

OBJ   VER      TOTAL      RIGHT
  1     1         15          7

  1     2          9          6

  1     3          8          6

  1     4          9          8

  1     5         10          3

  1     6          5          5

OBJ   VER      TOTAL      RIGHT
  2     1         17         12

  2     2         22         20

  2     3         15         13

OBJ   VER      TOTAL      RIGHT
  3     1         28         24

  3     2         22         19

OBJ   VER      TOTAL      RIGHT
  4     1         15         15

  4     2         21         19

  4     3         10          9

OBJ   VER      TOTAL      RIGHT
  5     1         28         17

  5     2         30         19

OBJ   VER      TOTAL      RIGHT
  6     1         25         18

  6     2         25         20
```